

DOCUMENT RESUME

ED 434 903

TM 030 131

AUTHOR Buiu, Catalin
 TITLE Artificial Intelligence in Education--State of the Art and Perspectives. ZIFF Papiere 111.
 INSTITUTION Fern Univ., Hagen (Germany). Inst. for Research into Distance Education.
 ISSN ISSN-1435-9340
 PUB DATE 1999-05-00
 NOTE 103p.
 PUB TYPE Information Analyses (070)
 EDRS PRICE MF01/PC05 Plus Postage.
 DESCRIPTORS *Artificial Intelligence; *Computer Uses in Education; Distance Education; Elementary Secondary Education; Foreign Countries; Higher Education; *Intelligence; *Intelligent Tutoring Systems; Interaction; Theories; World Wide Web

ABSTRACT

This review contains an overview of past and present trends in the application of what is called "artificial intelligence" in traditional face-to-face education and in distance education. The reviewed trends are illustrated with examples of research projects and results throughout the world. The first section of the review discusses intelligence in general, describing classic and modern theories of intelligence. Definitions of artificial intelligence in education are presented, with attention to these areas: (1) intelligent tutoring systems; (2) interactive learning environments; (3) virtual environments for education; (4) computer-mediated communication; (5) intelligent agents for education; (6) the World Wide Web and education; and (7) intelligent interfaces for learning support systems. Perspectives for the future of artificial intelligence in education are discussed. Nine appendixes contain a discussion of the Internet, a list of Web search engines, and other supplemental information related to artificial intelligence applications. (Contains 17 figures and 3 tables.) (SLD)

 * Reproductions supplied by EDRS are the best that can be made *
 * from the original document. *



ZIFF PAPIERE 111

Catalin Buiu

Artificial Intelligence in Education - State of the Art and Perspectives

PERMISSION TO REPRODUCE AND
DISSEMINATE THIS MATERIAL HAS
BEEN GRANTED BY

H. Fritsch

TO THE EDUCATIONAL RESOURCES
INFORMATION CENTER (ERIC)

1

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION
CENTER (ERIC)

- ☒ This document has been reproduced as
received from the person or organization
originating it.
- ☐ Minor changes have been made to
improve reproduction quality.
- Points of view or opinions stated in this
document do not necessarily represent
official OERI position or policy.

BEST COPY AVAILABLE

Zentrales Institut für Fernstudienforschung
FernUniversität - Gesamthochschule - Hagen
Mai 1999

Catalin Buiu

„Politehnica“ University of Bucharest
Department of Automatic Control and Systems Engineering
Spl. Independentei 313
Bucharest 77206
Romania
email: cata@cib.pub.ro

ZIFF PAPIERE

ISSN 1435-9340

Herausgegeben von Helmut Fritsch

Redaktion: Frank Doerfert, Helmut Fritsch, Helmut Lehner (Konstanz)

©1999 Zentrales Institut für Fernstudienforschung

FernUniversität -Gesamthochschule -
D 58084 Hagen
<http://www.fernuni-hagen.de/ZIFF>

Tel.: +49 2331 9872592
Fax +49 2331 880637

Artificial Intelligence in Education - State of the Art and Perspectives

by

Catalin Buiu

CONTENTS

List of figures	4
List of tables	4
1. Introduction	5
2. Intelligence, cognitive science and consciousness. Definitions and discussions	7
3. Teaching and learning. Artificial Intelligence in education	17
3.1 Reflections on teaching and learning. From face-to-face education to distance education	17
3.2 Artificial Intelligence in Education	23
4.1 Intelligent Tutoring Systems	29
4.2 Interactive Learning Environments	38
4.3 Virtual Environments for Education	41
4.4 Computer Mediated Communication	48
4.5 Intelligent Agents for Education	51
4.6 WWW and education	60
4.7 Intelligent interfaces for learning support systems	61
5. Future perspectives	66
Appendixes	69
Assignments	92
References	102

List of figures

Figure 1	Distributed artificial intelligence.....	10
Figure 2	Components of a learning environment.....	17
Figure 3	Components of a learning environment (part 2).....	18
Figure 4	Anatomy of an ITS.....	31
Figure 5	An intelligent tutor for basic algebra.....	36
Figure 6	An ILE: a systems theory point of view.....	38
Figure 7	POLYGONS microworld.....	39
Figure 8	A typical MUD database interaction.....	43
Figure 9	A definition for an agent.....	52
Figure 10	Categories of content in MOO (a) and whiteboard (b) interactions.....	59
Figure 11	WWW browser.....	60
Figure 12	Adaptive human computer interface.....	63
Figure 13	Belvedere system – a graphical interface for critical inquiry.....	64
Figure 14	Collaborative problem solving experimental setup.....	65
Figure 15	The structure of the collaborative problem solving system.....	86
Figure 16	The evolution of facts number in case of HercuLOG solving the problem alone.....	88
Figure 17	A Tcl/Tk interface for collaborative problem solving.....	89

List of tables

Table 1	Differences between Artificial Intelligence and Emergent Intelligence.....	11
Table 2	Benefits of Agent Technology.....	55
Table 3	The Main Cycle.....	87

1. Introduction

Education and learning are key factors in developing a new society for a new millenium. The new economies of the next century will become highly knowledge intensive. We will face the need of a rapid change in the work force. This, of course, will require an efficient effective training of the personnel.

The need for highly educated and trained people is more and more acute nowadays. There is a big increase in demand for learning from adults who must learn continously to stay current in the workforce, eventually by moving to a new carreer if necessary. We can notice, even from our own experience, that there is an increasing demand for a flexible learning framework, as most of the studying population has additional obbligations – mainly family and work. We live in an era of lifelong learning (Beller and Or, 1998). Hanna (1998) claims that: „throughout the industrial era, the system has focused upon serving the educational needs of youth to prepare for a lifetime of work. Today it is clear that the future will involve a lifetime of learning in order to work.“ Throughout the world, the post-secondary learning market has become one of today's growing markets, both in developed nations (Ghazi & Irani, 1997) and in developing ones. Various new models of higher education are evolving in North America and around the globe, in response to lifelong learning needs and to the new opportunities that are becoming available through the integration of learning technologies (Beller and Or, 1998).

The educational background of a man looking for a job is the first point that is taken into account. Of course, there are also other criteria for selecting the best suited applicant, such as creativity, originality, intelligence, memory etc. Although for the most people these attributes of a highly trained man seem to be native, the truth is that all of them can be developed through a process of education and self-training. Learning how to learn becomes more and more important than the traditional learning process, i.e. that of simply learning facts and rules. We can even say that this traditional learning cannot be named „learning“ anymore, as it does not involve an active and living presence of the human subject that is learning, which, in our oppinion, is a key factor of true learning.

So, everyone has the chances to improve his (her) own intellectual, psychological and spiritual performances. People have to be educated about the fact that this possibility really exists and they have to use the best techniques and methodologies for improving their overall performances which sometimes are telling more, and consequently are much more important, than a rigid educational background.

We live in an era of digital competition. That means that the world is facing rapid and impressive developments that are fueled by rapid advances in digital communications, digital satellite technologies, worldwide web, virtual reality, computer and learning technologies. Anyone of us cannot imagine, for example, a world without Internet.

The aim of designing better techniques and implementing new technologies for education is the driving force for the field of AIED. Although this name can be a little bit confusing because of the presence of a very controversial notion, that of „intelligence,, the idea behind it is to use the most performant techniques for improving the quality of the educational process and, a very important issue, for reducing the cost of education. The cost of education is a very important matter and there are individuals who claim that „Thirty years from now, the big university campuses will be relics. Universities won't survive... Do you realize that the cost of higher education has risen as fast as the cost of health care? Such totally uncontrollable expenditures, without any visible improvement in

either the content or the quality of education, means that the system is rapidly becoming untenable. Higher education is in deep crisis. Already we are beginning to deliver more lectures and classes off-campus via satellite or two-way video at a fraction of the cost. The college won't survive as a residential institution.“ (Forbes Magazine, 1997, pp. 126-127).

The field of AIED is quite fractured and there are many points of view of what it means. The main goal of this paper is to present all these points of view, to discuss them, to select the essential and to give an unitary point of view of the field of AIED.

We will give a personal overview of past and present trends in the applications of what is called „Artificial Intelligence“ in traditional face-to-face education and in distance education. We do not have the ambition to survey all the key issues and ideas in AIED. All the reviewed trends are exemplified by research projects and results worldwide. We will discuss also about perspectives projecting to potential future applications in the field. As we have already said, the field is fractured and very rich in notions and approaches, but our approach will be to reduce it to some essential features that mark the development of this research area. We will present these features, together with some examples and up-to-date applications, including our personal work in the field. Each issue discussed will be followed by some exercises, questions, themes for reflection that will be proposed to the reader. It is our hope that the reader will follow our argumentation with interest and that he (she) will actively take part to it, as we try to make this paper a living entity that would eventually give birth to interesting discussions and, why not, to new ideas for future developments in the field of AIED.

Our review is organized as follows. The first section will present some definitions and discussions about intelligence. We will review some classic and modern theories of intelligence, together with some ancient approaches to the nature of intelligence and consciousness. In our opinion, discussing about the intelligence and setting up a clear point of view on what intelligence is and on what it is not represents the best starting point in our approach of reviewing the field of AIED.

Next, we will give some definitions of AIED, we will analyze them and we will extract the essential of them so that we will be able to come with an unitary point of view. Accordingly to this proposed unitary view, we will discuss about each key area of AIED in respective sections.

Finally, we will discuss about AIED perspectives.

2. Intelligence, cognitive science and consciousness. Definitions and discussions

"It often does more harm than good to force definitions on things we don't understand. Besides, only in logic and mathematics do definitions ever capture concepts perfectly. The things we deal with in practical life are usually too complicated to be represented by neat, compact expressions. Especially when it comes to understanding minds, we still know so little that we can't be sure our ideas about psychology are even aimed in the right directions. In any case, one must not mistake defining things for knowing what they are."

Marvin Minsky, from „The Society Of Mind,, 1985

Intelligence

„Everything in the computer world called <intelligent> is by definition NOT.“

Intelligence is defined by the *Random House Dictionary* as:

1. capacity for learning, reasoning, and understanding; aptitude in grasping truths, relationships, facts, meanings, etc.
2. mental alertness or quickness of understanding.
3. manifestation of a high mental capacity: He writes with intelligence and wit.
4. the faculty or act of understanding.

A recent text defines AI to be the „study of the computations that make it possible to perceive, reason, and act“ (Winston, 1992). If the perception, reasoning, and action of this definition are related to human agents in a more immediate sense, the question arises as to why that should be so. Can't we speak of intelligence in the context of non-human agents? (Kak, 1996).

In „Unified Theories of Cognition“, Allen Newell defines intelligence as: „the degree to which a system approximates a knowledge-level system.“ A system is said to be a knowledge-level system when it rationally brings to bear all its knowledge onto every problem it attempts to solve. Thus, knowledge is the medium of transaction at the knowledge level and the behavioural law is the principle of maximum rationality.

Perfect intelligence is defined as the ability to bring all the knowledge a system has at its disposal to bear in the solution of a problem (which is synonymous with goal achievement). Here we note that we can speak of intelligence in connection with a goal or a task to be solved. Intelligence has to be distinguished from ignorance, a lack of knowledge about a given problem space.

The Turing Test was proposed by Alan Turing in 1950; he called it the Imitation Game. In 1991 Hugh Loebner started the Loebner Prize competition, offering a \$100,000 prize to the author of the first computer program to pass an unrestricted Turing test. Annual competitions are held each year with smaller prizes for the best program on a restricted Turing test.

The well known Turing test proposes the following protocol to check if a computer can think (or it is intelligent):

- (1) The computer together with a human subject are to communicate, in an impersonal fashion, from a remote location with an interrogator;
- (2) The human subject answers truthfully while the computer is allowed to lie to try to convince the interrogator that it is the human subject.

If in the course of a series of such tests the interrogator is unable to identify the real human subject in any consistent fashion then the computer is deemed to have passed the test of being able to think. It is assumed that the computer is so programmed that it is mimicking the abilities of humans. In other words, it is responding in a manner that does not give away the computer's superior performance at repetitive tasks and numerical calculations.

Martin Fischler and Oscar Firschein (1987) describe three philosophical theories on intelligence, theories they call, „existence theories“:

1. Intelligence is a nonphysical property of living organisms, and cannot be re-created in a machine.
2. Intelligence is an emergent property of organic matter: silicon is inadequate, but when we learn how to build machines out of organic compounds, we might have a chance of inducing intelligent behavior.
3. Intelligence is a functional property of formal systems, and is completely independent of any physical embodiment.

The first existence theory is based on dualism. Most generally, the dualism is the view that reality consists of two disparate parts; thus the mental is at least not identical with the physical. The mind and body are two distinct entities. The mind can be seen as the human awareness and consciousness. From this point of view, intelligence is tied to spirituality, which is purely philosophical. Thus, it would be impossible to re-create intelligence on a machine, unless by accident.

The second existence theory identifies intelligence as a property of evolved organic systems. Many Artificial Intelligence (AI) scientists are representing neurons in computer systems. The neural networks of the human brain are almost too complex to comprehend: let alone re-create it in silicon microprocessors. However, technology is advancing. Molecular biologists are beginning to use DNA molecules to solve complex computational problems (Baum 1996, p.57).

The third existence theory identifies intelligence as a property of formal systems. This is the viewpoint most computer scientists are concerned with.

Fischler and Firschein have broken down the idea of intelligence into attributes of an intelligent agent (1987):

1. have mental attitudes (beliefs, desires, and intentions)
2. learn (ability to acquire new knowledge)
3. solve problems, including the ability to break complex problems into simpler parts
4. understand, including the ability to make sense out of ambiguous or contradictory information
5. plan and predict the consequences of contemplated actions, including the ability to compare and evaluate alternatives
6. know the limits of its knowledge and abilities
7. draw distinctions between situations despite similarities
8. be original, synthesize new concepts and ideas, and acquire and employ analogies
9. generalize (find a common pattern in superficially distinct situations).
10. perceive and model the external world
11. understand and use language and related symbolic tools

Artificial intelligence

A possible very simple definition might say that AI is an set of information technologies which exhibit some ability for intelligent behaviour via artificial means.

Although it is quite difficult to define AI more rigorously, it is very obvious that the study of artificial intelligence has provided better programming techniques for building smarter computer systems.

From the Newell's point of view (described above), Artificial Intelligence could be viewed as the application of artificial or non-naturally occurring systems that use the knowledge-level to achieve goals. A more practical definition that has been used for AI is attempting to build artificial systems that will perform better on tasks that humans currently do better. Thus, at present, tasks like real number division are not AI because computers easily do that task better (faster with less error) than humans. However, visual perception is AI since it has proved very difficult to get computers to perform even basic tasks. Obviously, this definition changes over time but it does capture the essential nature of AI questions.

A brief history of AI would help us in understanding more of its research areas and characteristic approaches. This is taken from (Gams, 1995) and it is divided in 6 periods. Each one is presented in the following together with the dominant ideas of that period.

1. Early Enthusiasm or Tabula Rasa Craze (1955-1965):

- human brains are several orders of magnitude slower than computers; therefore, making a copy of a human brain on a computer would have to result in something more performant.
- 3 predominant subjects:
 - learning without knowledge;
 - self-organising systems and decision space techniques;
 - evolutionary learning.

2. Dark Ages (1965-1975):

- First approaches produced no fruitful results and new approaches are searched for;
- This era recognizes that „to acquire knowledge one needs knowledge“.

3. Renaissance (1975-1980)

- taking modest aims more appropriate to the level of technology produced even better results than expected;
- characteristics:
 - 1) exploration of different strategies;
 - 2) knowledge intensive approaches;
 - 3) successful applications;
 - 4) conferences and workshops worldwide.

4. AI Boom (1980-1990)

- expert systems are an example of commercial boom;
- superprojects are in progress (like the Japanese Fifth Generation Computer Systems).

5. New AI Winter (1990-1995)

- Superprojects produce no commercially successful programs;
- Overexpectations backfired again and criticism emerged with two basic claims:
 - 1) intelligence cannot be easily achieved on existing computers with existing approaches and methodologies;
 - 2) computers and approaches are not fundamentally differing from those of 30 years ago.

6. Invisible AI (1995-...). First Dawn Approaching?

- AI produces working systems although it has dissappeared from the first pages;
- AI techniques are invisibly interwoven with existing systems.

Several extensions to classical AI are available. For example, we can talk about DAI as it is, as we will see, involved in designing intelligent systems for education.

The field of Distributed Computing has existed ever since it became possible to use more than one processor to work on a computing problem. Originally, these processors shared the data associated with a given problem. The majority of issues that arose pertained to the parallelization and synchronization of the different processors. Soon after people started thinking about distributed computing, AI researchers became involved, trying to add automated intelligence to the field. AI efforts examined the prospect of sharing control as well as data among processors. The resulting research became known as DAI. It was distinct from traditional Distributed Computing in that it focussed on problem solving, communication, and coordination as opposed to the lower level parallelization and synchronization issues.

An introductory definition says that „Distributed Artificial Intelligence (DAI) is concerned with the study and design of systems consisting of several interacting entities which are logically and often spatially distributed and in some sense can be called autonomous and intelligent.“ (Weiß, 1995).

Other definition says that DAI studies „how a loosely coupled network of problem solvers can work together to solve problems that are beyond their individual capabilities“ (Durfee et al., p. 85).

Because of the emphasis on information management in the past, DAI has been decomposed into Parallel AI and Distributed Expert Systems, the latter of which was broken into Distributed Knowledge Sources and Distributed Problem Solving (DPS). However, these days people tend to break DAI systems into one of two classes: DPS and Multi-Agent Systems (MAS) (see Fig. 1).

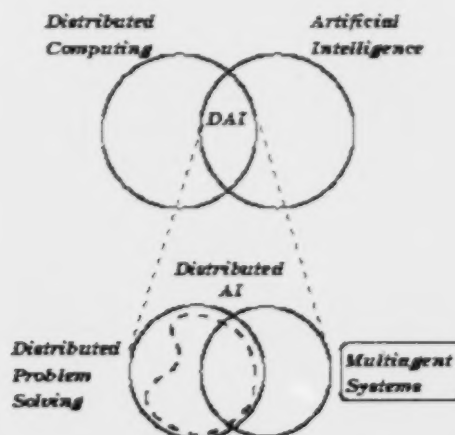


Figure 1 Distributed artificial intelligence

DAI is the intersection of Distributed Computing and AI. DAI consists of MAS and „not MAS“, most of the latter of which are DPS systems (but not all, as indicated by the dotted curve).

Emergent Intelligence vs. Artificial Intelligence

Traditionally, problem solving methods in artificial intelligence (AI) were divided in two categories: *weak* and *strong*. A strong method is containing a significant amount of problem-specific knowledge. A weak method is using little or no knowledge about the task. As their name says, the strong methods are more powerful algorithms than the weak methods.

There are cases of powerful weak methods for problem solving, and an example could be that of Genetic Algorithms. Speaking about GAs (or evolutionary algorithms, in general), it is important to note that additional knowledge about the problem to be solved is often available and it is possible to exploit problem specific knowledge in every step of a GA (Grefenstette, 1987). Let us describe briefly how an EA works. An EA is combining exploration of the search space with exploitation of the potential solutions. This is why EA may be called an *adaptive* weak method. So, an adaptive weak method differs fundamentally from a standard weak method. An adaptive weak method is becoming increasingly informed during the search and this informedness is the key difference from a standard weak method. So, as it is said in (Angeline, 1993) it is the interaction of the adaptive weak method with the problem environment that allows knowledge to *emerge*.

There has been much interest in the last years to study new paradigms of computation, such as connectionist models, classifier systems, artificial life, cellular automata, which have common features. Summarizing these features, Forrest defined *emergent computation* (EC) as having three parts (Forrest, 1990):

1. a collection of agents, each following explicit instructions;
2. interactions between agents (according to the instructions), which form implicit global patterns at the macroscopic level, i.e., epiphenomena;
3. a natural interpretation of epiphenomena as computations.

Following this idea of emergent computation, Angeline introduced the notion of *emergent intelligence* in (Angeline, 1993):

„Emergent intelligence (EI) is any problem solving algorithm that uses emergence as an integral component of the problem solving process such that the amount of explicit knowledge required to implement the same problem solving ability is reduced“

The task environment implicitly contains all knowledge associated with the task. From this observation, it comes the difference between EC and EI. From the above definition of EC, it results that there must be a local interaction between agents in order that EC result. In EI, there is also an interaction of agents with the task environment.

A summary of the comparison between EI and AI is taken from (Angeline, 1993) and presented in Table 1.

Property	Artificial intelligence	Emergent intelligence
task – specific knowledge	integrated into problem solver	separate from problem solver
Operators	task – specific	representation - specific
processing	centralized and global	local and opportunistic
knowledge representation	Explicit	emergent
knowledge content	Static	dynamic and adaptive
credit assignment	task - specific and static	representation specific, empirical and dynamical

Table 1 Differences between Artificial Intelligence and Emergent Intelligence

Artificial life (AL) models are an example of the reliance of emergence as a computational feature. AL is an example of systems with EI. Artificial life (alife) is the bottom-up study of basic phenomena commonly associated with living agents, such as self-replication, parasitism, evolution, competition, and cooperation. Alife complements the traditional biological and social sciences concerned with the analytical and experimental study of living organisms by attempting to synthesize life-like behavior within computers, robots, and other man-made media. One goal is to enhance the descriptive and predictive understanding of life both as-we-know-it and as-it-could-be. A second goal is to use nature as an inspiration for developing algorithms to solve difficult optimization problems exhibiting high-dimensional search domains, nonlinearities, and/or multiple local optima. According to (Tessier, 1994), AL is trying to synthesize the basic phenomena that are associated with living agents within computers or robots. Among these phenomena there are evolution, self-replication, self-organization, adaptation, co-operation, parasitism, and competition. Thus, AL is complementing the traditional biological and social sciences concerned with the analytical and experimental study of living organisms.

Cognitive science

We are interested in CS as it supplies the „fuel“ for the AIED field, that are detailed cognitive models and task analyses for ever more sophisticated kinds of reasoning and problem solving. This research has advanced from beginnings where only well-defined and closed worlds like logic, puzzles, games, and algebra were understood (e.g., Newell and Simon, 1972). Today, for example, we have information processing models of problem solving in knowledge-rich subjects like medicine (e.g., Clancey, 1987), physics (e.g., Larkin, 1980), and electronics troubleshooting (e.g., Frederiksen, White, Collins and Eggan, 1988) (McArthur, Lewis and Bishay, 1996).

A brief discussion on CS will start with what „cognitive“ means. We, as humans, think and learn. These are cognitive (or mental) abilities. Among our cognitive skills are the following:

- We can speak and understand languages
- We can reason and use logic to solve problems or discuss matters
- We can see things, recognize objects and understand the meaning of what we see
- We can synthesize different points of view in an unitary one
- We can form mental images in our head
- We can design, invent, create...

Basically, „Cognitive Science“ is the scientific study of our cognitive skills. This study is performed in the following ways:

- We do psychological experiments on people to see how their minds work - how quickly they can solve problems, how changes in the body and brain are related to mental changes, how people differ in their responses and how they are similar.
- We use computers to imitate or simulate human cognitive skills and to test theories about how the human mind works.
- We learn about human brains, about the different functions performed by different parts of the brain, about changes in behavior that are caused by damage to the brain.
- We learn about how these cognitive abilities are used in working with computers and other machines.

The knowledge we acquire by performing these experiments can and has to be applied in real life:

- In the development of computer systems with human capabilities, such as speech, speech recognition or even vision.

- In the development of machine hardware or software which is designed for people to understand easily and work with efficiently (such as user-friendly programs).
- In writing software, particularly communications, multimedia applications, virtual reality and other applications requiring computer intelligence with an understanding of human cognitive abilities.
- In medical research, architecture, engineering and numerous other fields which are benefiting from the new virtual reality worlds that allow professionals to explore new environments, such as the inside of the heart or the inside of an as-yet-unbuilt building.
- In designing new learning paradigms.

Taking into consideration Newell's theory of intelligence, hierarchically, the knowledge level lies above the symbol level where all of the knowledge in the system is represented. In other words, a tacit assumption in AI is that a knowledge-level system must contain a symbol system (i.e. this is simply a re-statement of the physical symbol system hypothesis). According to Newell's unified theory of cognition, the symbol level corresponds to the cognitive band and the knowledge level to the rational band. However, in the latter case, humans only approximate a knowledge-level system, due to the constraints imposed by bounded rationality. In „Unified Theories of Cognition,, Newell suggested that psychology is the study of this approximation, when the architecture „shows through“ and behaviour is not mediated by knowledge alone.

Distributed cognition

The Distributed Cognition approach was developed by Ed Hutchins and his colleagues at University California, San Diego in the mid to late 80s as a radically new paradigm for rethinking all domains of cognitive phenomena (Rogers, 1997). The traditional view of cognition is that it is a localised phenomenon that is best explained in terms of information processing at the level of the individual. In contrast Hutchins was making the claim that cognition is better understood as a distributed phenomenon.

The theoretical and methodological base of the distributed cognition approach derives from the cognitive sciences, cognitive anthropology and the social sciences. The distributed cognition approach uses a number of methods: from detailed analysis of video and audio recordings of real life events, to neural network simulations and laboratory experiments. The type of methodology adopted depends on the unit of analysis that is being adopted and the level at which the cognitive system is being explained.

A general assumption of the distributed cognition approach is that cognitive systems consisting of more than one individual have cognitive properties that differ from those individuals that participate in those systems. (Rogers, 1997)

The same author says that : „Distributed Cognition is a hybrid approach to studying all aspects of cognition, from a cognitive, social and organisational perspective. The most well known level of analysis is to account for complex socially distributed cognitive activities, of which a diversity of technological artefacts and other tools and representations are an indispensable part.“

Distributed cognition is based on the idea that cognition is distributed over the person and the person's environment. It combines the two extreme position of cognitivism and situationism. Cognitivism claims that all cognition happens in a person's mind and nowhere else, whereas situationism, in its most extreme form, claims that all cognition happens in a person's environment. Although there are some scientists who take one of the extreme stances, most are somewhere inbetween the extreme positions.

The relationship between social interactions and individual cognition issue lies at the very heart of the „social versus individual“ debate that concerns the nature of cognition. One can discriminate three theoretical positions with respect to this issue (Dillenbourg, 1992).

1. Piaget and the socio-constructivist approach (Doise and Mugny, 1984) are often presented as the defender of the individual position. Constructivism, the cognitive theory, was invented by Jean Piaget. His idea was that knowledge is constructed by the user. There was a prevalent idea at the time (and perhaps today as well) that knowledge is transmitted, that the student was copying the ideas read or heard in lecture directly into his or her mind. Piaget theorized that that's not true. Instead, learning is the compilation of complex knowledge structures. The student must consciously think about trying to derive meaning, and through that effort, meaning is constructed through the knowledge structures. Piaget liked to emphasize learning through play, but the basic cognitive theory of constructivism certainly supports learning through lecture -- as long as that basic construction of meaning takes place.
2. Vygotsky and the socio-cultural approach are viewed as the instigator of the social approach. As Butterworth (1982) pointed out, their opposition has been exaggerated. Both authors acknowledge the intertwined social and individual aspects of development, but they attribute the primacy to the individual (for Piaget) or to the social environment (for Vygotsky).
3. Distributed cognition. In this approach, the similarities between the individual and the social planes of cognition receive more attention than their differences. They view cognition as fundamentally 'shared' or 'distributed' over individuals.

Constructionism is more of an educational method which is based on the constructivist learning theory. Constructionism, invented by Seymour Papert who was a student of Piaget's, says that learning occurs „most felicitously“ when constructing a public artifact „whether a sand castle on the beach or a theory of the universe.“ (Quotes from his chapter „Situating Constructionism“ in the book „Constructionism“ edited by Papert and Idit Harel.)

The concept of distributed cognition emphasizes the interaction among individual, environment, and cultural artifacts. Oshima, Bereiter, and Scardamalia (1995) based on distributed cognition, examines students in knowledge construction and transforming in CSILE network environments. The results indicated that students who benefited most from the activities, engaged more in knowledge-transformation. This system allows students to distribute information and interact with information resources in a joint space, can prompt conceptual progress (knowledge assimilation and knowledge construction). Dede (1996) predicts a distributed learning and knowledge-building community will be the new paradigm of 21st century education.

A general assumption of the distributed cognition approach is that cognitive systems consisting of more than one individual have cognitive properties that differ from those individuals that participate in those systems.

Another property is that the knowledge possessed by members of the cognitive system is both highly variable and redundant. Individuals working together on a collaborative task are likely to possess different kinds of knowledge and so will engage in interactions that will allow them to pool the various resources to accomplish their tasks. In addition much knowledge is shared by the individuals, which enables them to adopt various communicative practices (e.g. not having to spell out every time they meet someone what they know about a practice, procedure or state of affairs).

Another important property is the distribution of access to information in the cognitive system. Sharing access and knowledge enables the coordination of expectations to emerge which in turn form the basis of coordinated action.

Metacognition

Metacognition refers to those cognitive processes that have the person's own cognitive processes as object. Thinking about what one is currently doing by stepping back and somehow looking over one's own shoulder is an important metacognitive activity (see reflection on current activity, reflection on past and reflection on future). Another important metacognitive activity is to ask oneself the right questions. (Obviously, asking somebody else a question is not a metacognitive activity.)

Consciousness

Consciousness is for the most of researchers the biggest mystery in the Universe. Some consider it the „largest outstanding obstacle in our quest for a scientific understanding of the Universe.“ (Chalmers, 1996). There are many attempts to explain it and to give a theory on consciousness.

I do not want to review all these approaches, but to outline the basis of an ancient theory on Consciousness that is interesting for the AI and CS communities as well. It was developed by the non-dual school of Kashmir Saivism in India more than 1000 years ago. It is interesting to note important similarities of this spiritual system with the modern works of quantum physics, e.g. the theories developed by David Bohm (Implicate Order).

The ultimate reality in Kashmir Saivism is called Shiva, the Supreme and transcendental Consciousness. The chief characteristic of Shiva, or the Self, is that Shiva is the Absolute. „Absolute“ is used by the Indian sages with two meanings. Firstly, it means which is independent, i.e. that is sustained by Itself. Shiva is both self-existent and self-illuminated (self-known). Secondly, „absolute“ means that which covers and pervades all. Everything in the Universe is a self-manifestation of Shiva.

All the questions related to the nature of Consciousness were considered by the Shaivite philosophers of India who further developed the earlier Vedic ideas (Abhinavagupta 1989, Dyczkowski 1987). Ordinary consciousness is bound by cognitive categories related to conditioned behavior.

According to the ancient doctrine of Samkhya, reality may be represented in terms of twenty five categories. These categories form the substratum of the classification in Shaivism. These categories are:

- i) five elements of materiality, represented by earth, water, fire, air, ether;
- ii) five subtle elements, represented by smell, taste, form, touch, sound;
- iii) five organs of action, represented by reproduction, excretion, locomotion, grasping, speech;
- iv) five organs of cognition, related to smell, taste, vision, touch, hearing;
- v) three internal organs, being mind, ego, and intellect; and inherent nature (prakriti), and consciousness (purusha).

But this classification is not rich enough to describe the processes of consciousness as it is mentioned as a single category (Kak, 1996).

Kashmir Shaivism enumerates further characteristics of consciousness (Abhinavagupta, 1989):

- vi) sheaths or limitations of consciousness, being time, space, selectivity, awareness, creativity, self-forgetting, and
- vii) five principles of the universal experience, which are correlation in the universal experience, identification of the universal, the principle of being, the principle of negation and potentialization (Shakti), and pure awareness by Itself (Shiva).

The first twenty five categories relate to an everyday classification of reality where the initial five characteristics relate to the physical inanimate world, and the next eighteen define the characteristics of the conscious organism. The inherent nature of the individual is called prakriti while purusa represents self. The next eleven categories characterize different aspects of consciousness which is to be understood in a sense different to that of mental capacities (categories 21,22,23). One of these mental capacities is akin to artificial intelligence of current computer science, which is geared to finding patterns and deciding between hypotheses. On the other hand categories 26 through 36 deal with interrelationships in space and time between these patterns and deeper levels of comprehension and awareness. (Kak, 1996).

The cognitive categories of Shaivism are of relevance in artificial intelligence (Kak, 1996). At present only a subset of these categories can be dealt with by the most versatile computing machines. Current research is focused on the lower categories such as endowing machines with action capacities (as in robotics) and powers of sense perception (as in vision). At the higher levels, while machines can be endowed with some capacity for judgment that typically involves computation of suitably framed cost functions, or finding patterns, or choosing between hypotheses, the capacities of concretization and especially self-awareness seem to be completely out of the realm of present day computing science. As regards sheaths of consciousness, constraints due to time, space, and selectivity are incorporated in many computers by pre-selecting the data to be processed or by tuning in the sensors to specified time windows. In general even the most elementary constraints require active intervention by human agents.

3. Teaching and learning. Artificial Intelligence in education

3.1 Reflections on teaching and learning. From face-to-face education to distance education

„Learning is finding out what you already know.

Doing is demonstrating that you know it.

Teaching is reminding others that they know just as well as you.

You are all learners, doers, teachers.“

Richard Bach, „Illusions“

First considerations

In discussing the role of technological support in education, Sandberg, (1994, p. 225) identifies the components of a (technologically rich) learning environment. These components must all be there in order to optimize learning (Fig. 2). However, they can be „implemented“ in many different ways.

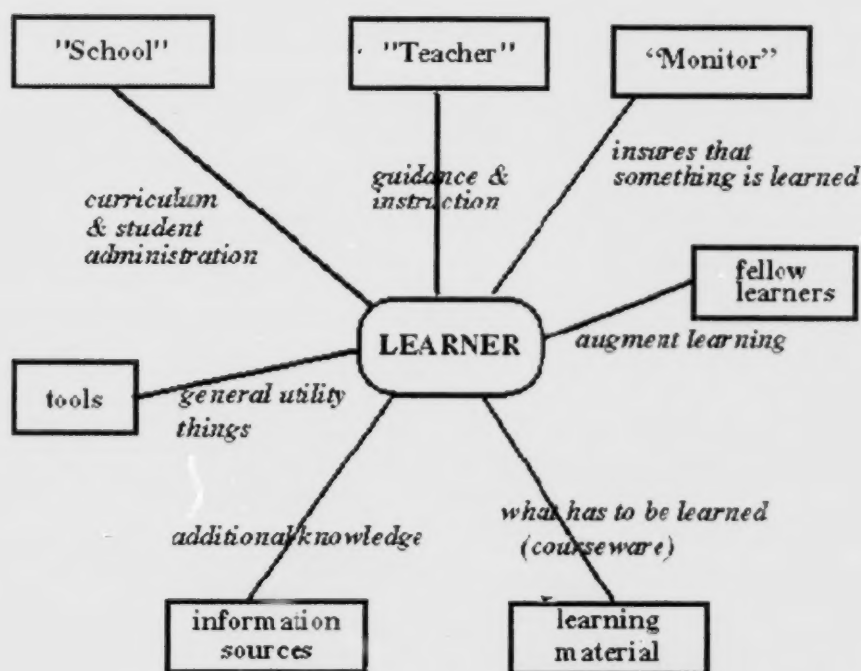


Figure 2 Components of a learning environment

1. "Teacher" component: Its role is to provide something between loose guidance and direct instruction. It can be a human agent (present or distant), an intelligent agent, instructions like some text books provide, etc. This component provides information from the syllabus to the task level.
2. "Monitor" component: Ensures that something is learned. A role taken by either the human teacher, the learner (self-control) or by some program.
3. "Fellow learners" component: Improves the learning process (some research tries to implement artificial ones).

4. „Learning material“: Contains what has to be learned in a very broad sense (knowing what, knowing how). It can be computational in various ways (exploratory hypertext, lesson and task oriented hypertext, simulation software, task solving environments, etc.).
5. „External information sources“: All kinds of information which is not directly stored in the learning material (e.g. additional material, handbooks, manuals, etc.).
6. „Tools“: Everything which may help the learning process other than the learning material (e.g. calculators, communication software, etc.)
7. „School“ [a category added by Schneider (1997)]: Something that provides a curriculum and does student administration.

At a different level, one can identify the following three key components: (1) Information, (2) Communication, and (3) Computer-based learning (CBL) (see Fig. 3).

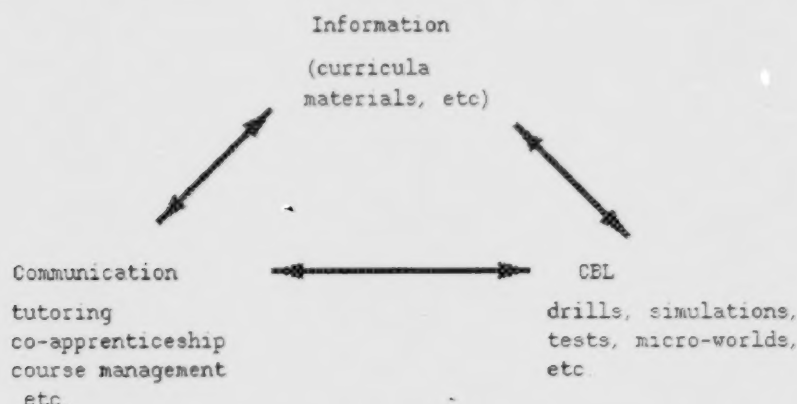


Figure 3 Components of a learning environment (part 2)

They typically come in separate form, Information and Communication being pre-dominant in the Internet World.

Let's have a closer look at „learning“ and „teaching“ (Schneider and Block, 1995):

What is learning? In short:

1. One learns by doing something (psychology)
2. One learns by pursuing an instructional goal (education).

When speaking about learning and teaching, we may take into consideration the next questions (Funderstanding website, 1999):

1. Learning theories: how do people learn? Here are some theories that aim at explaining how learning occurs:

- Constructivism
- Behaviorism
- Piaget's Developmental Theory
- Neuroscience
- Brain-Based Learning
- Learning Styles
- Multiple Intelligences
- Right Brain/Left Brain Thinking
- Communities of Practice

- Control Theory
- Problem-Based Learning

According to (Kearsley, 1993) there are different types of learning:

Attitudes: „.....Disposition or tendency to respond positively or negatively towards a certain thing (idea, object, person, situation).“ Also: Choose to behave this or that way according to opinions and beliefs.

Factual Information (Memorization): Processing of factual information and remembering is tied to previous knowledge. Memory research has also a lot to say about processing constraints.

Concepts (Discrimination): Concept learning encompasses learning how to discriminate and categorize things (with critical attributes). It also involves recall of instances, integration of new examples and sub-categorization. Concept formation is not related to simple recall, it must be constructed.

Reasoning (Inference, Deduction): „Reasoning encompasses all thinking activities that involve making or testing inferences. This includes inductive reasoning (i.e., concept formation) and deductive reasoning (i.e., logical argument). Reasoning is also closely related to problem-solving and creative behaviors“.

Procedure Learning: Procedures refer to being able to solve a certain task by applying a procedure. Once a procedure is mastered its excused usually does not take much effort (e.g. ftp a file). Cognitive theories like *Act* or *Soar* are interested in this, because procedures are important in diminishing cognitive load.

Problem-Solving: A good example is Newell and Simon's information processing paradigm for the study of problem-solving and the concepts of „means-ends-analysis“ and „problem space“. According to their GPS framework, problem-solving involves the identification of subgoals and the use of methods (especially heuristics) to satisfy the subgoals. And important contribution was also the methodology of protocol analysis (of „thinking aloud methods“ which has been extensively used by Anderson (1987) to implement intelligent tutoring systems according to his *Act** theory (Anderson, 1983).

Learning Strategies: can be learned too to some extent. Very much dependant on what you want to learn

We may note that learning types can be strongly related to different kinds of cognitive task behaviors (that are being used while learning or that are targets for learning). As an example, Kearsley (93) lists the following types of task behaviors:

- Searching for/receiving information (detects, observes, inspects, identifies, reads, surveys)
- Processing information (categorizes, calculates, codes, itemizes, tabulates, translates)
- Problem-solving (analyzes, formulates, estimates, plans)
- Decision-making (examines, chooses, compares, evaluates)
- Communication (advises, answers, directs, informs, instructs, requests, transmits)
- Sensory-motor processes (activates, adjusts, connects, regulates, tracks)

2. **Curriculum**: what should be learned? The answer is given by:

- Outcome-Based Education (OBE)
- Core Curriculum
- Whole Language
- Character Education
- Multiculturalism
- Tech-Prep
- Paideia

3. **Instruction**: how should learning be designed? Here are some possible answers:

- Mastery Learning
- Cooperative Learning
- Accelerated Learning
- Thematic Instruction
- Whole-Brain Teaching
- Service Learning
- Cognitive Coaching
- School-to-Work Transition
- Instructional Technology
- Youth Apprenticeship

4. **Assessment**: how will we know if learning occurs? By using:

- Authentic Assessment
- Classroom Assessment Techniques
- Portfolio Assessment

5. **Organizational Theory**: how should schools be designed? As:

- Total Quality Schools
- Charter Schools
- Accelerated Schools
- Comer Schools
- Coalition of Essential Schools

Teaching can be characterized by 2 aspects:

(1) Teaching „setups“: e.g. Distance Teaching, Open Learning, Semi-Distance Teaching, Traditional Class room teaching, etc. Each of those setups needs different instructional strategies and tactics.

(2) Instructional processes that can be looked at from many points of view. Let's have a short look at normative instructional design theory: What is the optimal sequencing of course-ware and how is it related to various types of learning?

For many educational technologists „course-ware“ is the production of computerized learning materials that would fall into one of the following categories (Schneider and Block, 1995):

1. Programmed Instruction (transfer of content proceeds step-by-step)
2. Computer Assisted Instruction (Drills and Tutorials)
3. Intelligent Computer Assisted Instruction (ITS Tutorials)
4. Computer Based Learning (Simulations, Hypertext and Micro-worlds)
5. Intelligent Learning Environments (Micro-worlds + tutors, helpers, experts)
6. Cognitive Learning Support Environments (some hypertexts)
7. Knowledge Construction Environments.

Gagné suggests nine universal steps of instruction (cf. [Gagné, 1985] or [Aronson et Briggs, 1983]) which should be found in any instructional context:

1. Gain attention: e.g. present a good problem, a new situation, use a multimedia advertisement.
2. Describe the goal: e.g. describe the goal of a lesson (task,...), state what students will be able to accomplish and how they will be able to use the knowledge, give a demonstration if appropriate.
3. Stimulate recall of prior knowledge: e.g. remind the student of prior knowledge relevant to the current lesson (facts, rules, procedures or skills). Show how knowledge is connected, provide the student with a framework that helps learning and remembering. Tests can be included.
4. Present the material to be learned: e.g. text, graphics, simulations, figures, pictures, sound, etc. e.g. follow a consistent presentation style, chunking of information (avoid memory overload, recall information)
5. Provide guidance for learning: e.g. presentation of content is different from instructions on how to learn. Should be simpler and easier than content. Use of different channel.
6. Elicit performance: „practice“, let the learner do something with the newly acquired behavior, practice skills or apply knowledge
7. Provide informative feedback: show correctness of the trainee's response, analyze learner's behavior (or let him do it), maybe present a good (step-by-step) solution of the problem
8. Assess performance: test, if the lesson has been learned. Also give information on general progress
9. Enhance retention and transfer: inform the learner about similar problem situations, provide additional practice. Put the learner in a transfer situation. Maybe let the learner review the lesson.

Instructional Design Theory provides a detailed prescription on how to organize teaching and learning at the global (curricula), lesson and task level. Most work is also grounded in some learning theory.

Distance education today

During the past decade there have been numerous efforts to reinvigorate distance education; some of these have involved video technology, but others have made use of computers and advances in the design of computer software. Distance education is the best example on how technological developments will affect the traditional learning strategies and systems that we will know. That justify In 1995, the State University of New York, funded by a major grant from the Sloan Foundation, embarked upon the creation of an „on-line“ learning program--a set of asynchronous and distance learning courses delivered solely via computer (Heath, 1998).

A first definition of Distance Learning (DL) can be the following:

„A learning intervention in which the students and/or instructors are not in the same physical space.“

Others are listed below:

„Distance Education is defined as a planned teaching/learning experience that uses a wide spectrum of technologies to reach learners at a distance and is designed to encourage learner interaction and certification of learning.“

--defined by University of Wisconsin-Extension, Continuing Education Extension, Distance Education subgroup

„Distance Education is instructional delivery that does not constrain the student to be physically present in the same location as the instructor. Historically, Distance Education meant

correspondence study. Today, audio, video, and computer technologies are more common delivery modes.“

--defined by Virginia Steiner. The Distance Learning Resource Network (DLRN)

„Distance education (or correspondence/home study) is the enrollment and study with an educational institution which provides lesson materials prepared in a sequential and logical order for study by students on their own. When each lesson is completed the student makes available, by fax, mail, or computer, the assigned work for correction, grading, comment, and subject matter guidance by qualified instructors. Corrected assignments are returned to the student, an exchange which provides a personalized student-teacher relationship.“

--defined by The Distance Education and Training Council (DETIC)

„The term distance education represents a variety of educational models that have in common the physical separation of the faculty member and some or all of the students.“

--from A Conceptual Planning Tool, developed by the University of Maryland System Institute for Distance Education

„At its most basic level, distance education takes place when a teacher and student(s) are separated by physical distance, and technology (i.e., voice, video, data, and print), often in concert with face-to-face communication, is used to bridge the instructional gap.“

--by the Engineering Outreach staff at the University of Idaho; from the Guide:Distance Education at a Glance

Using the first definition, all of the following would be examples of DL (WWW6, 1997):

- Satellite-based delivery of a class, where the instructor is in New Jersey and the students are at a single location in Florida.
- Satellite-based delivery of a class, where the instructor is in New Jersey and the students are scattered at multiple locations world-wide, including some co-located in a classroom with the instructor.
- Delivery of a class over a learning network, where students receive the learning at their own workstations in their offices.
- Broadcast-based delivery of course material over cable TV, with mail-in, paper-based or computer-based course materials.

To this definition, the participants to (WWW6, 1997) agreed that Internet-Based Distance Learning offers several extensions of capability:

- It can accommodate both synchronous and asynchronous learning, often within a single „course.“
- It can be interactive or non-interactive. Examples of non-interactive would be single-user access to static course materials.
- It can deliver a rich blend of media within a single learning intervention. In other words, this single medium encompasses the range of functionality and interactivity that used to require multiple channels or media to achieve.
- The infrastructure for accessing Internet-Based DL is widely available and cross-platform.
- The same environment can be used for learning and performing the actual task or transaction.
- One can incorporate teaching materials into your learning intervention that you have not created for himself, and that have been authored by people he has never met.
- Students can publish their own work, thereby increasing motivation and expanding the scope of their capabilities. This may have the added effect of encouraging greater depth or care in their research and writing processes, as the audience for their publication is so much wider than would be possible otherwise.

3.2 Artificial Intelligence in Education

As we live the Internet era, all of us are becoming more or less Internet minded and oriented. We haven't been an exception to this rule when we started to write this review. Our first approach was to search the web with „AIED“ as a keyword (see Appendix 2 for a list of search engines). The results were unexpectedly scarce. Why? A first explanation could be that AIED is just a nice term coined by few researchers for a lot of approaches and researches undertaken in improving the quality of education and learning. AIED seems to be an heterogenous field trying to bring together researchers from a lot of individual research areas.

We will follow the (few) results of our Internet search and we will give some definitions, discussing upon the main concepts.

A. Journals

One of the first results says that there exists an „International Journal of Artificial Intelligence in Education (IJAIED)“. Let's see what it is the scope of this Journal: „AIED is concerned with the application of artificial intelligence techniques and concepts to the design of systems to support learning“

B. Conferences

Next, we noticed that there are even International Conferences on AIED. Let's check for some more details.

1. *European conference on AI in education*, Lisbon, Portugal, September 30 - October 2, 1996

„EuroAIED is intended to provide an opportunity for researchers to develop principles for the design of systems to support learning. The „principles“ may be derived from artificial intelligence, cognitive science, human-computer interaction, and related fields; the „systems“ include tutoring systems, learning environments, simulations, multimedia systems, WWW-based systems, etc.; and „learning“ includes learning by schoolchildren, university students and industrial trainees.“

Here we notice the same general point of view as before. AIED is concerned with designing „systems to support learning“. They explicitly mention that AI is one of the fields that inspire researchers in order to improve the quality of learning.

But let us check the workshops organised for that Conference.

- | | |
|---------------------------|--|
| 1. Virtual Reality | 11. External Representations |
| 2. Collaborative Learning | 12. Natural Language |
| 3. Distributed Cognition | 13. Self-explanation |
| 4. Machine Learning | 14. Internet-Based Learning Environments |
| 5. Motivation | 15. Educational Multimedia |
| 6. Qualitative Reasoning | 16. Social Learning |
| 7. Student Modelling | 17. Instructional Planning |
| 8. Cognitive Science | 18. Simulation-based learning |
| 9. Authoring Systems | 19. Other Innovations |
| 10. Metacognition | |

We can notice some old research areas within AI such as machine learning, natural language or qualitative reasoning. There are also some „learning“ related areas (student modelling, learning

environments). Cognitive science is mentioned explicitly as it is the area that is trying to give detailed models of human cognition (ability to learn being one of the most important human cognitive ability). Let's take now a more global view and consult some world conferences.

2. AI-ED 95, 7th World Conference on Artificial Intelligence in Education
Washington DC, USA, 1995

They say nothing about what AIED means but they gave major session topics:

- | | |
|---|---|
| • Intelligent tutoring systems | • Conceptual change/Metacognition |
| • Learning environments and microworlds | • Teaching higher-order thinking skills |
| • Visual and graphical interfaces | • Social and cultural aspects of learning |
| • Human factors and interface design | • Cognitive development and errors |
| • Non-standard and innovative interfaces | • Student modeling, cognitive diagnosis |
| • Intelligent multimedia systems | • Theories of teaching/Motivation |
| • Authoring systems and tutoring shells | • Reading and writing |
| • Collaboration tools | • Educational robotics |
| • Principles/tools for instructional design | • Computer-assisted language learning |
| • Natural language interfaces | • Evaluation of instructional systems |
| • Knowledge representation for instruction | • Assessment of learning outcomes |
| • Knowledge and skill acquisition | |

Here we note the presence of ITSs, the veteran of AIED field.

Let's move now from U.S. to Japan.

3. Workshop on Current Trends and Applications of Artificial Intelligence in Education,
in conjunction with The fourth world congress on expert systems,
Victoria, Australia, December 9 - 13, 1997

„The workshop focuses on the current trends, perspectives and applications of Artificial Intelligence to the development of educational software that supports learning in our global society. This workshop provides an opportunity to meet and discuss artificial intelligence techniques for student modelling, agents that support learning, language learning, virtual reality and educational software in the WWW.“

Here AI is viewed as one of the tools to be used for developing high performance educational systems. New trends are explicitly mentioned, such as agents and virtual reality. The connection with AI is quite weak and it will grow weaker as the time will pass.

4. AI-ED 97, 8th World Conference on Artificial Intelligence in Education,
Kobe, Japan, August 1997

„The theme for 1997 will be Knowledge and Media in Learning Systems, and papers that explore the emerging roles of intelligent multimedia and distributed technologies as well as computer supported collaboration within that theme will be particularly welcome.“

The technical program focuses on research activities linking Artificial Intelligence theories and techniques with Educational theory and practice. Areas of interest include but are not limited to:

- Agents
- Architectures
- Authoring systems and tutoring shells
- Case-based systems
- Cognitive development
- Cognitive diagnosis
- Collaboration and collaborative tools
- Computer-assisted language learning
- Conceptual change
- Educational robotics
- Evaluation of instructional systems
- Human factors and interface design
- Intelligent multimedia and hypermedia systems
- Intelligent tutoring systems
- Knowledge and skill acquisition
- Knowledge representation for instruction
- Learning environments and microworlds
- Modelling pedagogical interactions
- Motivation
- Natural language interfaces
- Networked learning and teaching systems
- Non-standard and innovative interfaces
- Principles and tools for instructional design
- Social and cultural aspects of learning
- Student modelling
- Higher-order thinking skills and metacognition
- Theories of teaching
- Visual and graphical interfaces.“

Comparing with the previous list of topics we may notice a new research area, that of „Agents“. ITS is no more present in the list of topics and actually there were only two papers related to ITS that were presented at the Conference.

And now back to Europe:

5. AI-ED 99, 9th World Conference on Artificial Intelligence in Education,
Le Mans, France, July 1999

They give no topics of interest, but a very generous theme: „Open learning environments: Towards new technologies to support learning, exploration and collaboration“.

Here we notice a return to a more pragmatic approach that insist on exploring new technologies for improving learning.

Taking this pragmatic point of view we refined our Internet search, not insisting anymore on „Artificial Intelligence“ keywords, but on „information technology and education.“

Here's an example of what we have found:

6. CAL 99, 29 – 31 March 1999, The Institute of Education, London, UK

Its main theme is „Virtuality In Education: What are the future educational contexts?

A radical effect of the combination of multimedia and pervasive networks has been the emergence of virtuality. What are the implications of virtuality for learning and teaching? In particular, what are the implications of emerging notions of virtuality which conceive virtual representations as part of the real world in themselves. For example, how will cyber-societies, which are viewed as no less real by their participants than local neighbourhoods, impact on learning and teaching. What are the questions that educators need to debate for the future in which virtuality is commonplace?“

CAL99 has focussed on the following topics:

- The changing role of the learner
- Continuing professional development
- Theories of learning and virtuality
- Learning communities
- New forms of educational software
- Virtuality and the curriculum
- Learning in public
- Virtualities and institutions
- Learning in the home
- The changing role of the teacher
- Learning at work
- Identity in virtual environments

We may notice here the explosive development of an area of AIED, that of virtual reality for learning and education. We can say that most of the traditional areas of AIED, especially those very technologically oriented, have become very wide research areas nowadays.

6. *ICCE 99, 7th International Conference on Computers in Education* - New Human Abilities for the Networked Society - November 4 - 7, 1999. Chiba, Japan

The conference theme, „New Human Abilities for the Networked Society“, indicates a focus on the new education needed in the near-future societies that are rapidly developing in the region. New technologies including the Internet, multimedia, communication satellites and artificial intelligence open up exciting possibilities. Educational communication and collaboration are becoming regional and even global. At the same time the effectiveness of school and industry education must be improved, and continuing education expanded. The challenge is to find the best ways to exploit technology to enhance the creativity, collaboration, and communication that will be at the heart of the new education for the next century.“

Let's browse the topics:

- Agent Architecture
- Architectures for Educational Technology Systems.
- Artificial Intelligence and Networked Learning.
- Cognitive Modeling.
- Collaboration. (university/school/industry)
- Computer Supported Collaborative Learning.
- Contents Production with Authoring Systems.
- Creativity.
- Cross Cultural Education.
- Data Mining Technology in Education.
- Distance Education.
- Educational CALS.
- Educational Simulation and Gaming.
- Electronic Commerce in Education.
- Electronic Manual Writing.
- Global Education.
- Human Competence.
- Human Development in Industry.
- Improving Classroom Teaching.
- Industry Training System.
- Instructional Design.
- Interactive Learning Environments.
- Internet, Web Application.
- Intelligent Tutoring Systems.
- IT Education.
- Language Education.
- Life Long Education/Learning.
- Measurement and Evaluation.
- Modeling Tools and Activities in Education.
- Network and Hypermedia Navigation.
- Networked Social Learning Models.
- Ontology in Education.
- Pedagogical Agent.
- Pedagogical Foundations.
- Performance Supporting System.
- Policy, Ethics, Standards, and Legal Issues.
- Satellite Education.
- Social and Cultural Issues.
- Sociological and Cultural Impacts of the Internet.
- System Design and Development Methodologies.
- System Evaluation.
- Teaching/Learning Strategies.
- Teacher Training System.
- Technology Transfer.
- Teleconferencing.
- Video On Demand for Education.
- Virtual Reality.
- Web-Based ITS/CAI.

Here AI is explicitly mentioned and this Conference points the revival of true AI used in education.

C. Academic research groups

Here is a summary of key research areas for AIED taken from an important academic research unit that is TECFA in Switzerland (<http://tecfa.unige.ch>):

1. Help systems & Tools for learning:
 - Models of Learning & Computational Metaphors
 - Help Systems for Learning
 - Hypermedia & Learning.
2. Distributed Cognitive Systems
 - Human Computer Interaction
 - Computer Supported Collaborative Learning
 - Learning & Collaboration with Artificial Agents.
3. Mediated Educational Communication
 - Pragmatics in Educational Communication
 - Semiotics in Educational Technology Products
 - Distance Education.
4. Information & Communication Systems
 - Multi-User Virtual Worlds
 - Virtual Campuses & Tools for Distance Education
 - Interactive Web Pages.

D. Human experts

Let's see what the experts are saying.

Schneider and Jermann definition (1997):

"Artificial Intelligence and Education (AI&Ed) [is a] community which over the last years tried to combine principles derived from artificial intelligence, cognitive science, human-computer interaction, and related fields. for the design of systems to support learning."

McArthur, Lewis and Bishay (1997):

„The hallmark of AI applications in education is that they attempt to explicitly represent some of the reasoning skills and knowledge of expert practitioners, and to exploit that expertise for teaching and learning.“

In (Jermann and Schneider, 1998) we have found the first reference to the age of AIED research: „25 years of AI&Ed research has taught us that the environment and the learner(s) are best described and designed as a single cognitive system and that we should focus on making the global system (that includes the learners) more intelligent.“

Another one is mentioned in Andriessen and Sandberg (1999): „For about 20 years the Intelligent Tutoring Systems paradigm has dominated the field of AIED“.

In the following they say: „We present arguments supported by research examples for a fundamental shift of emphasis in education and its relation to technology, in particular AI-technology. No longer the ITS-paradigm dominates the field of AI and Education. New educational and pedagogic paradigms are being proposed and investigated, stressing the importance of learning how to learn instead of merely learning domain facts and rules of application. New uses of technology accompany this shift.“

Mizoguchi, Ikeda and Sinita (1997): „Artificial intelligence in education (AI-ED) research consists of the following three major research areas in addition to Computer Science: Artificial Intelligence techniques and theories;

1. Artificial intelligence;
2. Educational technology;
3. Cognitive science.“

and:

„A lot of research on AI-ED has been done to date. Several learning paradigms such as CAI, ICAI, Micro-world, ITS, ILE and CSCL have been proposed and many systems have been built within each paradigm. The AI-ED community is strongly affected by innovative computer technologies such as hyper-/multi-media, virtual reality, Internet, WWW.“

The authors note that some AI technologies have successfully been introduced into AI-ED community to bring fruitful results in building various IESs (Intelligent-Educational Systems). However, the interaction between AI-ED and AI communities has been not so active compared to those with Educational technology and Cognitive science communities.

The same renowned expert, Riichiro Mizoguchi says (1999):

„There is no definition on what AIED research is, I am afraid. But, I think there is a consensus about we need „modeling“ in some sense. The term AI should be interpreted weaker, that is, the research doesn't have to use AI technology explicitly. But, it needs modeling and a sophisticated philosophy in system design.“

Let us summarize what we have found so far.

1. The main goal of AIED research is to design high performance artificial systems that support learning.
2. AIED is 20-25 years old and the first paradigm that dominated it was that of ITS. ITS area is concerned with the implementation of human teachers' intellectual abilities into the computer and it attempts to implement traditional methods of learning and teaching. Main AI topics were taken into consideration when an ITS was designed, such as knowledge representation, reasoning, natural language processing, explanation, machine learning and planning.
3. Spectacular technology improvements have changed the field during the 1980s and 1990s. New results proved that was possible to automate traditional methods of teaching and learning (McArthur, Lewis and Bishay, 1997). This has as a result the redefining of the educational goals. „Higher order' skills to be learned (such as creative mathematics, composing and writing) replace older „low order“ skills (such as symbol manipulation algebra and spelling correction (Ibidem). More results are achieved regarding „the acquisition of procedural skills, about predominant errors and misconceptions in specific domains, about domain viewpoints (representation), about the nature of fruitful dialogues between tutor and student.“ (Sandberg and Andriessen, 1999).
4. New technologies are booming during the mid 1990s. An era of digital competition has been born. Network communication, multi-media systems, virtual reality, Internet, WWW are concepts and practical tools that have significantly affected the AIED field. The number of conferences dedicated to ITS, the ancestor of the modern AIED systems, is dramatically decreasing. New theoretical results in cognitive science are providing information processing models of problem solving in knowledge rich subjects like medicine, electronics or physics, e.g. (Ploetzner, 1997). Consequently, educational technology and cognitive science communities have had important and active interactions with the AIED field than the field of AI. So, as Sanberg and Andriessen (1999) say: „The nature of the field [AIED] changed...but its name did not. Are we then not justified in asking the question where is AI and how about education?“

5. During the last 2-3 years, integration of AI techniques into systems that support learning is more active. One example is ontology engineering, an innovative research area in AI. This research is expected to build a new bridge between AI and AIED communities by overcoming the drawback of the current IES. For example, ontology makes an IES smart and reflexive and it explicates the conceptualization on which the system is based, it makes knowledge reusable and so forth (Mizoguchi, Ikeda and Sinitsa, 1997).
6. Nowadays, we can speak of the revival of true AIED area. Next generation learning support systems are just beginning to emerge as a result of a successful marriage between AI, computer science and educational technology. We can speak today or in the very near future about high performance ILE, about modelling of human high-level faculties and learning mechanisms, about CSCL and CSCW systems, information filtering systems that support on-line information searching, multi-agent systems for education, virtual environments for education, expert systems for course-ware design, WWW based CAI systems for distance education and so forth.

4.1 Intelligent Tutoring Systems

The first (and still foremost) application of AI to education has been to build „intelligent tutoring systems“ (ITS) (McArthur, Bishay and Lewis, 1997). ITSs incorporate the most advanced performance measurement systems with learning models from the field of cognitive psychology and software design methods from the field of artificial intelligence. This approach to computerized training provides students with their own optimum learning environment, simulating the one-on-one teaching methods of an expert human tutor.

ITS (or advanced automated instructional systems as they are often called) (Regian & Shute, 1992), differ from traditional computer-based instruction in two ways. More advanced systems often use 1) a software simulation of the target task context to provide the student with interactive practice opportunities, and 2) pedagogical software agents to make run-time curriculum decisions and manage student learning activities.

ITS have been under development at least since WEST (Burton and Brown, 1982) and SOPHIE (Brown, Burton and deKleer, 1982) nearly twenty years ago. Many specific systems, their structure and goals, have been discussed in depth by Sleeman and Brown (1982), Wenger (1987), and Psotka, Massey, and Mutter (1988), among others. Ohlsson (1986) and Schank and Edelson (1990) have also contributed valuable critical reviews of the direction of the field.

ITS and Artificial Intelligence

The „I“ in ITS is viewed from different prospects depending on the background of each researcher. What is interesting to search for is an effective use of „Artificial Intelligence“ in the design and development of ITSs with regard to unsolved problems at the current state of the art. There are many aspects of AI relevant of ITSs, i.e. symbolic artificial intelligence, neural networks, knowledge representation, reasoning, natural language processing, user interfaces, knowledge engineering, student modeling, cognitive issues, etc. Related topics include Intelligent Training and Coaching Systems.

With the use of AI, ITS can identify a student's strengths, weaknesses and preferred style of learning. These systems are able to qualitatively process information, recognize patterns of behavior, identify misconceptions or „bugs“ in performance, and establish a plan of instruction. Instruction can be tailored to the student's learning style and remediation is based on student errors and the „computer perceived“ misconceptions.

Expert systems, a major topic of AI, provide the framework to code, organize, and retrieve knowledge about a specific content domain. The heart of an ITS is its expert system. The expert system embeds sufficient knowledge of a particular topic area to provide „ideal“ answers to questions, correct not only in the final result but in each of the smaller intermediate reasoning steps. The expert system thus allows the ITS to demonstrate or model a correct way of solving the problem. Often, like a human tutors, it can generate many different answer paths or goal structures (McArthur, Stasz, Hotta, Peter, and Burdorf, 1988). If a student needs an explanation of why or how an algebra ITS did a step in solving an equation, the system might first say that it used the distributive rule. If the student requested more justification, it could elaborate by describing the terms that were distributed and the arithmetic „cleanup“ steps that followed. Explanations thus turn expert systems from opaque „black box“ experts into inspectable „glass boxes“ (Foss, 1987).

The knowledge structure simulates the „expert“ problem solver. Expert systems provide to ITS the capability of reasoning, justifying, interpreting, predicting, diagnosing, monitoring, planning, and controlling student behavior. ITS models the „human“ problem solver and recognizes the human capability of using a variety of methods to solve problems.

Anatomy of an ITS

Here is the anatomy of an ITS (adapted from Dede, 1992 and presented in Fig. 4). ITS are knowledge-based educational and they have historically been large, self-contained programs with specialized platform requirements.

ANATOMY OF AN INTELLIGENT TUTORING SYSTEM

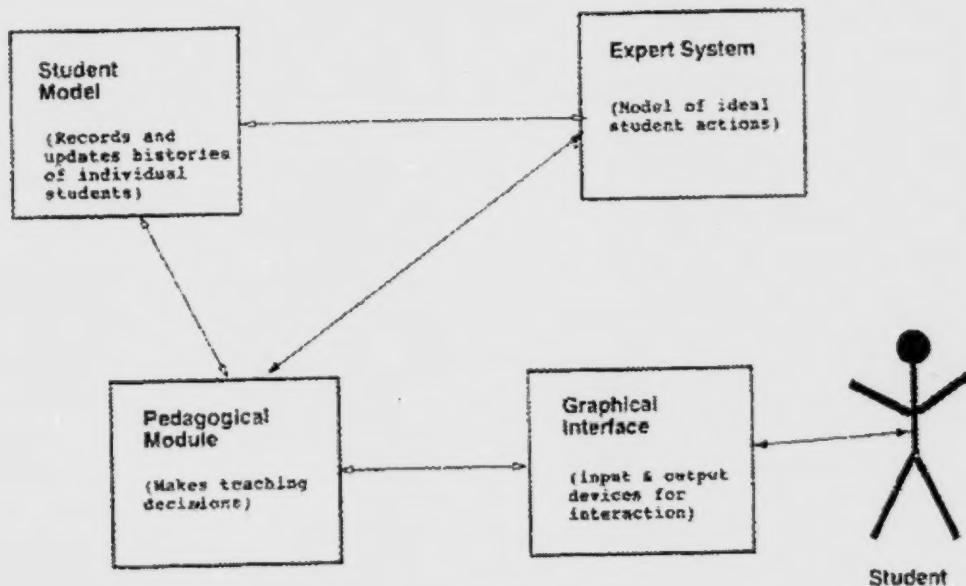


Figure 4 Anatomy of an ITS

The three major components of an Intelligent Tutoring System are the expert model, the student model and the instructional model.

The Expert Model

There are a variety of knowledge architectures (www.oti.navy.mil). The „black box“ model (Anderson, 1988) establishes a criterion referenced knowledge base. The content domain is naturally organized in an existing symbology which the computer understands. The computer can assess student performance without the need of „human intelligence“. The criterion for acceptable performance is clearly identified. If the input behavior does not meet the criterion, the computer will inform the student of the performance error and recommend possible solutions. The dialogue between the student and ITS is very simplistic. The computer does not provide detailed explanation of its reasoning. Mathematical equations lend themselves to this type of architecture.

Issue-based architectures compare the student input to the „expert“ model and the „student“ model. Instructions are programmed to specific issues which are observable in the behavior of the „expert“ model and the „student“ model. If the student's performance fails to meet the prescribed behavior criterion, the student receives immediate feedback. Feedback includes an explanation of the rule. The dialogue can be very superficial (i.e. simple feedback on the correct action) or very complex by providing detailed reasoning behind the behavioral rule.

Cognitive Model. Cognitive models realistically simulate human problem solving. The three levels of knowledge are procedural, declarative, and qualitative. Procedural Knowledge relates to how a task is performed. Declarative Knowledge is a set of facts organized to permit reasoning. The underlying premise of using this type of knowledge architecture is the assumption that the student has the

procedural knowledge base to make inferences from the content domain. Qualitative knowledge involves the causal understanding which allows humans to reason about behavior using the models of the system. Expert models using qualitative knowledge are in the developmental stage.

Knowledge is structured in a Production System (known as a rule-based system). A production system analyzes and synthesizes large quantities of knowledge to solve problems. Declarative and procedural knowledge are integrated in a production system. The declarative knowledge formulates a database of facts and the procedural knowledge provides the rule base in an IF-THEN or WHEN-THEN relationships (A rule which results in multiple actions is known as a WHEN-THEN relationship.) (Williams & Reynolds, 1990).

A rule is „a chunk of information which, when matched with conditions in the environment, triggers a specific behaviour (Williams & Reynolds, 1990, p. 28). „A rule consists of a set of facts and a set of actions. All conditions on the IF side of the rule are matched to a set of facts in the database. A set of behaviours are identified on the THEN side of the rule.

The production system has a controller. The controller searches the system and determines which rules are appropriate for the set of conditions. In situations where more than one rule applies to a set of conditions, the controller resolves the conflict and selects the most appropriate application to base its reasoning.

Intelligent Tutoring Systems have expanded the application of the production system by recognizing that humans have the capacity to solve problems using various reasoning methods. The IT systems assesses student performance to determine if the behaviour conforms to the prescribed „expert“ rule(s). If the performance does not match the rule, the ITS assumes the role of instructor, diagnoses the student's weaknesses and prescribes appropriate remediation. Intelligent Tutoring Systems also use meta rules to make inferences concerning what to teach independent of the content domain.

The Student Model

The student model establishes the framework for identifying the student's misconceptions and suboptimal performance. The structure of the student model can be derived from (1) the problem-solving behaviour of the student, (2) direct questions asked from the student, (3) historical data (based on assumptions of the student's assessment of his skill level, novice to expert), and (4) the difficulty level of the content domain (Barr & Feigenbaum, 1982).

ITS compares the student's actual performance to the „student model“ to determine if the student has mastered the content domain. Advancement through the curriculum is dependent upon the IT system's assessment of the proficiency level of the student. The student model contains a database of student misconceptions and missing conceptions. This database is known as the „bug library“. „A missing conception is an item of knowledge that the expert has but the student lacks. A misconception is an item of knowledge that the student has but the expert does not (VanLehn, 1988, p. 62). „Bugs are identified from the literature, observation of student behaviour and learning theory of the content domain (VanLehn, 1988).

The intelligent tutor solves problems similar to the human. The tutor predicts student performance. If the performance does not meet the prediction, the system must determine if the deficiency is due to a misconception or a missing conception. Once the tutor recognizes there is misconception or a missing conception, it makes a diagnoses and prescribes instructional remediation.

Instructional Model

ITS actively interacts to student inputs and diagnoses the student's level of understanding or misunderstanding of the knowledge domain. The tutorial exercises some control over the selection and sequencing of information by responding to student questions concerning the subject domain and in determining when the student needs help and what kind of help is needed (Halff, 1988). An effective ITS will meet the ever changing needs of the student. ITS diagnoses the student's characteristic weaknesses and adapts the instruction accordingly. As the student's level of proficiency increases, ITS will ideally conform to the evolving skill level of the student. ITS adapts as the novice evolves into a subject matter expert.

ITS Feedback Formats. Feedback increases the rate of acquisition as well as retention of learned behaviors (Williams & Reynolds, 1990, p. 23). ITS instructional feedback capabilities provide immediate knowledge of results. Burton (1988) identified seven formats for structuring instructional feedback to the student: help, assistance, empowering, reactive learning, modeling, coaching and tutoring.

Help Format. The HELP format allows the student to request assistance when they have made an error or when they perceive they need assistance. The on-line capability permits the student to learn by doing. The student perceives that he has control over the learning process. This capability has a positive impact on the student's acceptance of the system.

Assistance Format. In the Assistance format the intelligent tutor assumes some of the responsibility for problem solving tasks and allows the student to concentrate on specific areas. The IT system instructs the task by presenting task's operational sequence. The student is provided an opportunity to apply that operation and eventually generalize the operations in solving similar problems. This format facilitates the development of conceptual understanding and encourages higher-order thinking skills which are involved in problem solving (Burton, 1988).

Empowering Format. The Empowering format provides to the student the tools to review their own decision making processes. The system captures the student's performance decisions and their impact, and provides a visual representation of the student's problem-solving ability. The student travels through their own decision tree to identify errors made. Problem-solving behavior is acquired in a „risk free“ environment.

Reactive Learning Format. In the Reactive Learning format the IT system „responds to the student's actions in a manner that extends the student's understanding of their own actions in the context of a specific situation (Burton, 1988, p. 127).“ Initially, the student establishes a hypothesis which the computer challenges. The hypothesis is challenged on the basis of its logic, its compatibility with the information the student has previously learned, and its consistency with the knowledge base. The student is required to articulate and justify his/her own reasoning.

Modeling Format. The Modeling format models „expert“ performance for the student. The student learns by observing the „expert“ at work.

Coaching Format. The Coaching format simulates the „human“ coach. It constantly monitors student performance to identify suboptimal performance. ITS will immediately interrupt the interaction and provide advice to the student. The system compares the student's performance to its „expert“ model. If the student's performance deviates from the „expert“, the coach redirects the student towards the „expert“. The coaching format is not concerned with the student completing a

predetermined lesson. The primary emphasis is on skill acquisition and general problem-solving through computer games (Earr & Feigenbaum, 1982).

Tutor Format. The instructional „tutor“ identifies deficiencies in skill performance. The automatic instructional capability of ITS provides an environment to enhance learning. The instructional tutor identifies errors of commission, omission, and „bugs“ in student performance. ITS communicates through natural dialogue and provides remediation when necessary. The Tutor must determine when to interrupt and how often. Too little feedback or too much can hinder the learning process. ITS constantly analyzes the student's performance to ensure the learning process of the knowledge domain is being mastered.

The two types of instructional tutor formats are the expository tutor and the procedure tutor.

Expository Tutor. The Expository Tutor presents factual knowledge with an emphasis on the development of inferential skills (Halff, 1988). The expository tutor's primary mode of instruction is through a natural dialogue with the student. Factual knowledge is sequenced to provide a coherent structure to the learning process. The framework establishes a relationship between existing knowledge and general concepts. Instructional dialogue begins with generalities and proceeds to specifics.

Procedure Tutor. The Procedure Tutor provides instruction on procedural skills that can generalize to other situations. The procedure tutor emphasizes the development of effective problem-solving skills. ITS usually assumes the „coaching“ mode. Information is sequenced in the form of practice exercises and examples which are based on the student's accomplishment of specific instructional objectives. Guidance is provided throughout the learning process.

Applications and new trends

1. REACTT system (<http://www.csd.abdn.ac.uk/research/its.html>) attempts to learn co-operatively; that is, the system does not act as an „oracle“ but learns from the same experiments as the student. Although this is a very appealing approach, it remains to be shown empirically that these tutors are any more effective.

2. The INFER system (<http://www.csd.abdn.ac.uk/research/its.html>) is able to infer previously unencountered mal-rules from student protocols given background domain knowledge and a range of focusing heuristics.

3. COCA - A Co-operative Classroom Assistant

COCA contains a set of authoring tools for building simple ITSs and a tutoring shell. The teacher is able to build domain material, teaching strategies and meta-teaching strategies. COCA contains a general teaching strategy model which the teacher instantiates with different sets of rules. The meta-strategy gives control over how these strategy rules are refined during teaching. The system has been evaluated by reconstructing several teaching strategies from the literature using COCA, and then further in a case study with a secondary school mathematics teacher. A final evaluation was an empirical experiment comparing school and university teachers performing a set authoring task.

4. SHAI (<http://www.shai.com>)

SHAI is constructing and modeling an Intelligent Tutoring System (ITS) server which connects, and promotes communications between, a loose confederation of ITSs maintained by individuals with little or no knowledge of each other's existence over the Internet. A user interacts with one ITS. When it determines that the student lacks knowledge in a related field which is handled by another ITS, it sends him or her there. When these switches occur, the student model is also sent, so that the

other ITS instantly knows the state of the student's knowledge in related fields and learning style preferences. Additionally, SHAI is developing an ITS authoring tool that provides instructors and teachers a visual and integrated environment to create interactive multimedia ITSs which can be deployed and distributed via the internet.

SHAI is using the authoring tool to create an ITS for the US Air Force AWACS program. It will train Weapons Directors to improve their ability to understand the current tactical situation and to quickly, concisely, and clearly convey this to the fighter and other pilots. The ITS incorporates a variety of media and technologies, including graphics, video, sound, real-time animation and simulation, and speech recognition. More details can be found at <http://www.shai.com/projects/internet.htm>.

5. Applications at NASA

(http://groucho.gsfc.nasa.gov/Code_520/Code_522/Projects/ITS/). There are two major on-going activities involving intelligent tutoring systems (ITS), both of which apply knowledge-based technologies and graphics to various forms of training and operational support. The first application of ITS technology is to improve NASA command and control language training, and the second activity is the extension of the textual/graphical presentation to include animation, video and audio (i.e., hypermedia). The basic objectives of both efforts are to demonstrate the applicability of hypermedia and ITS technology as a training device and as a tool for intelligent information access and presentation in NASA mission operation environments.

In conjunction with the Control Center Systems Branch (Code 511), a prototype ITS for the Systems Test and Operations Language (STOL) for the Gamma Ray Observatory (GRO) operators was developed. The system uses a Macintosh graphical user interface with an expert system matching the STOL rules to the current state of the tutor and then presenting the associated actions or consequences back to the student.

6. RGI's ITS Work (<http://www.cts.com>)

RGI designed and developed an ITS to train military and civilian government personnel how to administer the Computerized Adaptive Testing version of the Armed Services Vocational Aptitude Battery (CAT-ASVAB). RGI's evaluation of the CAT-ASVAB ITS found that it accomplished in 3 to 5 hours of computerized instruction what previously required a trainer 32 hours of classroom time.

The primary objective of their research is to assess the improved effectiveness and utility of an ITS which employs coaching systems and information processing approaches to measurement. For this research effort, RGI is designing, developing, and evaluating four different versions of a lesson that teaches PC troubleshooting skills.

7. An Intelligent Tutoring System for Basic Algebra (McArthur, Bishay and Lewis, 1997):

The authors' algebra ITS helps students learn freshman level algebra, focusing on equation solving and symbol manipulation. In different versions of the tutor the student can solve problems using symbols, operations, or commands. In the „Symbols“ version, students input new equations that lead toward a solution. Students can input equations either by typing or by using electronic pencil. In „Operations“ the student issues requests like „add 30 to both sides of the equation“. And in „Commands“ the student specifies very high level goals, like „collect“ like terms. The different versions allow the student to focus on different levels in the rather complex hierarchy of reasoning skills that characterize even simple algebra. Students practice one level of decision-making, and the ITS takes care of other levels that the student is not concentrating on at a given time. For example, if

the student says „collect“ the ITS will figure out what „collect“ means in terms of operations, then also do the appropriate symbol manipulation.

In Figure 5 the student is using the operations version of the algebra tutor. The operations the student chooses are recorded in the right side window. The left side contains various menus that allow the student and ITS to converse. For example the student can move scroll display, can ask for help, can ask the ITS to do a step and to explain what it was doing, and can create their own problem or get easier or harder ones.

Menu (Left):

- INEQUALITIES
- Scroll Right
- Scroll Left
- Scroll Down
- Scroll Up
- My Answer Ok?
- Help Next Step
- Explain Your Step
- Login
- Erase Input
- Move Back
- Homework Problem
- Student Problem
- Easier Problem
- Harder Problem
- QUIT

Central Workspace:

$$\begin{aligned}
 &-2+7x < 5(-6+x)-4(-6+x) \\
 &-2+7x < -54+9x-4(-6+x) \quad -2+7x < 5(-6+x) \\
 &-2+7x < -54+9x-4(-6+x) \quad -2+7x < -30+5x \\
 &-2+7x < -30+9x-4x \quad -2+7x < -30+5x \\
 &-2+7x < -30+5x \\
 &-2+7x+30 < -30+5x+30 \\
 &28+7x < 5x \\
 &28 < 5x-7x \\
 &28 < -2x \\
 &\frac{28}{-2} > \frac{-2x}{-2} \\
 &-14 > x
 \end{aligned}$$

Operations Window (Right):

- simplify: Reduce terms to a simpler form, or do arithmetic
- /: Divide both sides by ?
- *: Multiply both sides by ?
- +: Add ? to both sides
- : Subtract ? from both sides
- distribute: Expand a * or / using the distributive rule
- collect: Collect "like" terms using the distributive rule

Bottom Section:

CONDITIONS: There is more than one x on one side of $-2+7x < 5(-6+x)-4(-6+x)$.

GOAL: A useful goal was to COLLECT the variables together.

OPERATION: To carry out this goal the variables $5(-6+x)-4(-6+x)$ were combined to make $(5-4)(-6+x)$.

Figure 5 An intelligent tutor for basic algebra

8. MEDIATE The Medical image Description And Training Environment

(<http://www.cogs.susx.ac.uk/lab/hct/mediate/medhome.html>) MEDIATE aims to supplement professional training in radiology by offering computer-based training and diagnostic assistance. In addition, it provides trainee radiologists with exposure to a large structured archive of images and associated expert descriptions indexed by lesion appearance, position and diagnosis. It is also a tool that help radiologists develop a „conceptual framework“ and appropriate description language for magnetic resonance (MR) images of the head suitable for a wide range of imaging sequences.

The starting point for this project was a detailed structured description language for MR images of the head, which has been developed together with a dedicated image archive of 1200 cases illustrating a variety of pathologies, each with associated clinical presentation information and structured image descriptions.

This archive and description database provide key image and statistical data for the development of a neuro-radiological tutoring environment, the MR-Tutor, in collaboration with the School of Cognitive and Computing Sciences, Sussex University. The MR-Tutor project has brought together expertise in cognitive modelling, human-computer interaction, intelligent tutoring systems design, medical imaging, and medical statistics and is a remarkable example of up-to-date multi-disciplinary

research related to ITS. A prototype of the MR-Tutor was developed in collaboration with neuroradiologists and implemented under the HipWorks/Poplog environment.

9. Authoring shells and ITSs

The Intelligent Computer-Assisted Training Testbeds (ICATT) program of the US Air Force addresses the need for more cost effective methods for building intelligent tutoring systems. ICATT is designed to remedy this situation by producing authoring shells which allow instructional developers and subject matter experts (SME) with little or no programming skills to rapidly develop, deliver, and maintain ITSs on personal computer class machines.

Here are some examples:

1. Microcomputer Intelligence for Technical Training (MITT) Writer. Instructional developers could use MITT Writer to produce ITSs to train tasks involving the troubleshooting of complex devices. During field trials of six ITSs built with MITT Writer, it was demonstrated that not only could instructors and SMEs with little or no programming experience build ITSs, they could learn how to use MITT Writer and produce one hour of troubleshooting training in an average of 136 hours.
2. Rapid ITS Development Shell (RIDES). RIDES is an authoring shell that provides answers to some of these issues by tightly linking instructional development and delivery with the simulation of a complex device or graphical visualizations of domain concepts. RIDES is designed for training environments where learner interaction with computer-based graphical models of complex devices or domains can significantly enhance learning.

Limitations of Intelligent Tutoring Systems

While ITS have been somewhat successful on a small scale, several problems must be overcome before they have widespread impact. Various authors (e.g., Wenger, 1987; Psotka, Massey, and Mutter, 1988) have discussed a wide range of limitations. Many of these challenges can be predictably factored by ITS component -- limitations associated with the expert system, student model, pedagogical component, and interface. From interviews with teachers it was found that one of the reasons for a low audience of ITS in the real classroom is that most ITSs teach according to a fixed teaching strategy, and do not allow teachers to alter the way in which material is taught.

Although the instructional effectiveness of ITSs has been demonstrated many times, the costs of developing and maintaining these systems have made them difficult for customers to afford.

While many computer aided instruction programs have proven to be effective, the technology of intelligent tutoring has not, generally speaking, produced useful programs. Currently, research is conducted using vastly different subject domains and student populations, but more research is certainly needed.

4.2 Interactive Learning Environments

A learning environment (LE) is the (virtual) place where „teaching“ and „learning“ meet. A LE is a piece of educational software in which the learner is „put“ into a problem solving situation. A very simple and common example is that of a flight simulator in which the learner (the future pilot) learns how to fly an airplane and does not answers questions on how to pilot an airplane (as in an usual ITS). The learner is assisted by a tutor which in our case is an experienced pilot. If there are also other components of the LE that are not human and assist the learner we call it an Intelligent Learning Environment (ILE). Usually these no-human components are software agents that are usually implemented by using AI techniques.

So, as (Dillenbourg et al., 1993) say: „The words intelligent learning environment are used for learning environments which include (1) a problem solving situation and (2) one or more agents that assist the learner in his task and monitor his learning.“

For making a distinction to ITS, we can mention that in an ILE:

1. the learner learns by constructing his own knowledge – in an ITS the learner is told by lecture;
2. the learner himself has the control and not the tutor. In an ILE the tutor is only a guide;
3. in an ILE the student generally receive unique feedback and information as a result of their interaction with the LE – in an ITS the tutor is responsible for tailoring the feedback.

Concluding these preliminary remarks, we present in Figure 6 the structure of an ILE.

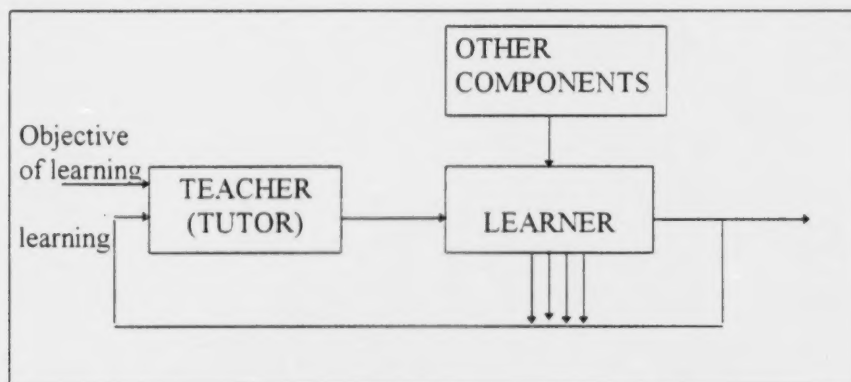


Figure 6 An ILE: a systems theory point of view

Microworlds are a particular kind of ILE. The microworlds could be stand-alone „construction kits“ for single users or teams of users, or located as pockets of activity within a networked multiuser environment (Strohecker, 1995). The design of the microworlds is such that exchanges of objects between the microworlds and of ideas among the users enhance the power of the kits as environments for learning. The microworlds described by Strohecker are for learning about an aspect of motion study, balance. They are also for learning about an aspect of topology, the spatial relationships between vertexes, edges, and faces of three-dimensional shapes. Users construct dinosaur skeletons and mobiles for experiments with balance, and polyhedra for explorations in topology.

Microworlds both move from tutors to tools and from a drill-and-practice method of teaching to an inquiry-based method of learning. Microworlds also represent a shift in the desired outcomes of learning. Many microworlds, have two distinct kinds of goals for student learning (McArthur, Bishay

and Lewis, 1997). First, as with ITS, it is usually important for students to learn subject-specific knowledge. More precisely, students often attempt to characterize the patterns of relationships among objects and properties that define the world. In SMITHTOWN (Shute and Glaser, 1990), for example, students might learn about the law of supply and demand by studying changes in costs of commodities as various factors influence supply or demand. Second, either implicitly or explicitly, many microworlds encourage students to learn inquiry skills themselves. These are relatively generic skills that students must know to conduct inquiries concerning virtually any topic. There is no rigorous task analysis of inquiry comparable to more „well-defined“ skills, like factoring quadratics or performing an integration. However, following Lakatos (1976) and Steen (1988), we have compiled what we believe is a consensus view of inquiry activities, summarized in Table 1. These inquiry skills have been championed in the past (e.g., Polya 1962) and more recently (e.g., Schoenfeld, 1985; National Council of Teachers of Mathematics, 1989). Proponents see these inquiry skills as intrinsically valuable „higher-order“ thinking abilities (Resnick, 1987). Thus, in microworlds inquiry is viewed both as a method of learning specific subject areas, and as a topic of learning.

Example 1 POLYGONS microworld

This is an example of a microworld for mathematical inquiry. The system is used by students (usually in pairs) with a mentor present to coach the students and occasionally to provide seed topics or issues. The role of the mentor fades over time as students become comfortable with the software and acquire better inquiry skills (McArthur, Bishay and Lewis, 1997).

In Figure 7 students are using Polygons to investigate the relationship of N and R (radius), as P (perimeter) is held constant. How is R changing (up or down) as N increases, and how fast?

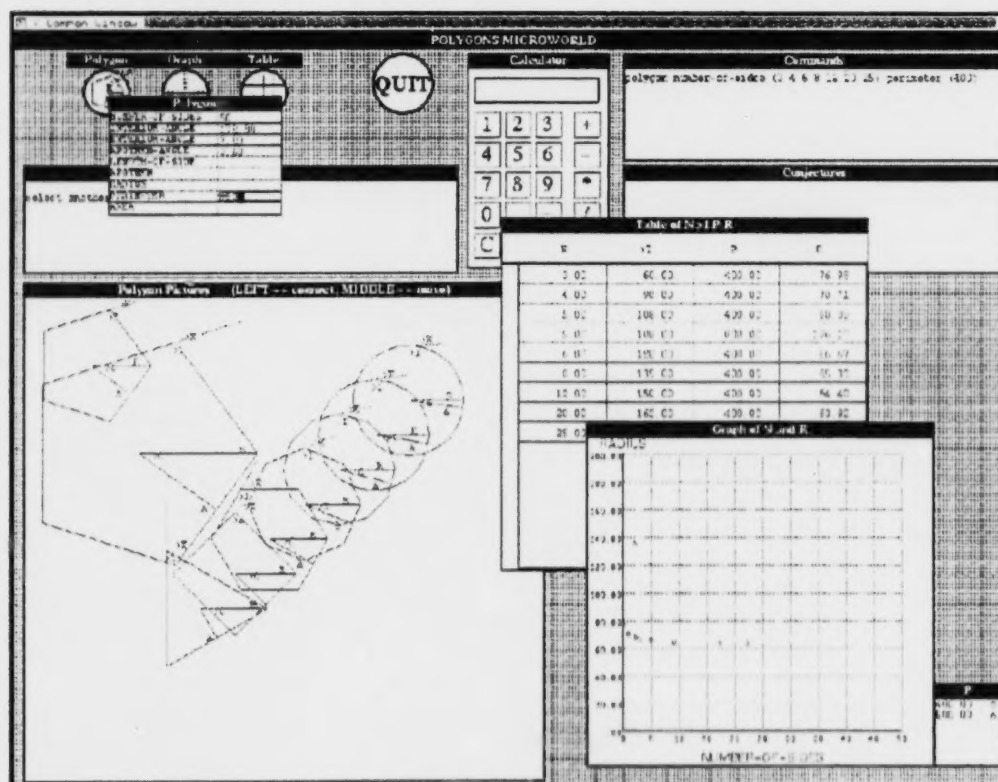


Figure 7 POLYGONS microworld

Students begin such investigations empirically, by generating several polygons (objects) and viewing them with different tools to find patterns. Objects are created in two way. First students can input values for certain properties in an „object table“, an instance of which can be seen in the upper-left corner of Figure 7. When two values are input the system computes all other values that logically follow, using equation solving and constraint propagation tools, borrowed from our the algebra tutor software. For example, if the student inputs an N of 8, then >X (exterior angle; 45 degrees), >I (interior angle; 135 degrees), and >A (apothem angle; 22.5 degrees) are computed automatically. If N and some scale parameter (e.g., P) are both given, then all other values are determined and displayed. The second way to create polygons is to issue commands that will make many at a time, using the commands window in the upper right. When polygons are created they appear on the large pictures window and can be manipulated in several ways: they can be moved, queried and connected into different representations, as we describe below.

Example 2. PEOPLE POWER (Dillenbourg, 1992a; Dillenbourg and Self, 1992), **MEMOLAB** (Dillenbourg et al., 1992; Dillenbourg and Mendelsohn, 1992) and **ETOILE** (Dillenbourg et al, 1993).

2.1 People Power

PEOPLE POWER is a learning environment in which the human learner interacts with an artificial learning companion („co-learner“). Its pedagogical goal is that the human learner discovers the mechanisms by which an electoral system is more or less proportional. The system includes four components:

1. a microworld in which the learner can design an electoral experiment (i.e. choose parties, candidates, laws, etc.), run the elections and analyze the results;
2. an interface by which the human learner (and conceptually the co-learner) plays with the microworld;
3. the co-learner, named Jerry Mander, and
4. an interface that allows the human and the computerized learners to communicate with each other.

2.2 MEMOLAB / ETOILE

The goal of MEMOLAB is for psychology students to acquire the basic skills in the methodology of experimentation. The learner builds an experiment on human memory. A typical experiment involves two groups of subjects each encoding a list of words. The two lists are different and these differences have an impact on the recall performance. An experiment is described by assembling events on a workbench. Then, the system simulates the experiment (by applying case-based reasoning techniques on data found in the literature). The learner can visualize the simulation results and perform an analysis of variance.

This artificial lab constitutes an instance of a microworld. Most learners need some external guidance to benefit from such a microworld. Computational agents (coach, tutors and experts) to provide this guidance. Another way of helping the learner is by structuring the world. MEMOLAB is actually a sequence of microworlds. The relationship between the objects and operators of two successive microworlds parallels the relationship between developmental stages in the neo-piagetian theory of Case (Dillenbourg and Mendelsohn, 1992). At the computational level, the relationship between successive worlds is encompassed in the interface: the language used in a microworld to describe the learner's work is used as the command language for the next microworld.

Advanced learning environments are difficult to implement and currently they are not cost effective. Now lets look at a few postulates that can be found in such research (Schneider and Block, 1995):

1. The Learner must be active;
2. A learning environment should be designed to be as powerful as dedicated working environments. It must be rich and complex reflecting the essential properties of what has to be learned;
3. The environment must be structured. If the richness of a learning environment is a quality, its complexity may reduce learning. It must provide optimal learning conditions as a function of the learner's stage of knowledge;
4. Learning environments should be designed as hierarchical knowledge base generators;
5. Learning environments should present knowledge as a communication system. A learner must interact with agents, tutors.

4.3 Virtual Environments for Education

An important issue in AIED research is that of Virtual Environments (VE). VEs for Education (VEE) are interactive cyberspaces where many users can communicate and collaborate in various way. They also can build virtual like offices, books, blackboards, artificial persons and more. VEs should also provide optimal support for information storage, retrieval and manipulation (Schneider and Godard, 1996).

VEEs for education, research and life have have been discussed in various disciplines. While advanced multi-user educational VEs are still mostly speculation, simpler VEs based on standard technologies have been in existence for some time. (e.g. [Eisenstadt et al., 1995]). With the emergence of new CSCW collaboration tools (e.g. [Kindberg, 1996] , the explosion of Web, renewed interest in collaboration research (computer supported collaborative learning, groupware, etc.), and billions invested into telematics and computer graphics VEs will become fashionable on a larger scale (Schneider and Godard, 1996).

A Collaborative Virtual Environment (CVE) is one that actively supports human-human communication in addition to human-machine communication and which uses a Virtual Environment (including textually based environments such as MUDs/MOOs) as the user interface. This is an exciting field with much potential for inter-disciplinary collaboration particularly in the fields of computer science, psychology, sociology, architecture and urban planning, cultural and media studies and Artificial Intelligence.

Many VE researchers stress the importance of collaboration and communication and experiment with currently available communication and information technology. A good example is the study of [Gay et Lentini, 1995] on the use of communication resources in a network collaborative design environment. On the other hand, graphical VR research has mostly focused more on the creation and manipulation of artificial 3-D worlds and the theoretical issue of immersion. Both approaches will probably soon meet on some middle ground but there will without doubt remain a large variety of interests and applications. In any case, VEs have great potential for education and collaborative work.

VEEs are often used as a complement to traditional distance learning. We can mention here the STAF (Sciences et Technologies de l'Apprentissage et de la Formation) Diplôm offered by the TECFA unit at the University of Geneva (Schneider, 1998). STAF uses a special semi-distance teaching format, in which there are 6 x 1 week of intensive courses in-site / year combined with independent study periods using the network. The tools that are used are the following:

- (1) WWW (hypertext), e.g. for:
 - planning, curricula, agendas;
 - texts, resources (and pointers);
 - student assignments (cooperation by imitation);
 - collaboration by group projects.
- (2) Email, e.g. for:
 - agenda planning (teacher);
 - search for information (student);
 - information about updates (student, teacher);
 - short comments (teacher).
- (3) Discussion Forums, e.g. for:
 - debates (about articles or themes);
 - technical questions and answers.
- (4) The virtual world itself (TecfaMOO), e.g. for:
 - urgent things;
 - co-presence (common virtual space, common radio channels);
 - virtual meetings;
 - non-intrusive (almost) real time questions/answers.

Background on MOOs

A MUD (Multi-User Dungeon or, sometimes, Multi-User Dimension) is a network-accessible, multi-participant, user-extensible virtual reality whose user interface is entirely textual. Participants (usually called players) have the appearance of being situated in an artificially-constructed place that also contains those other players who are connected at the same time. Each player can communicate easily with each other in real time.

The earliest MUDs were written in 1978-1979 and were based on the role-playing game Dungeons and Dragons. In 1989, a graduate student at Carnegie Mellon University named James Aspes decided to see what would happen if the monsters and magic swords were removed. He created a new type of MUD, called TinyMUD,“ which was not an adventure game. Later versions included a simple programming language. Instead of spending time killing virtual monsters, participants work together to help extend the virtual world. Langdon Winner remarks that „social activity is an ongoing process of world-making“ (Winner, 1986).

Here comes the Internet: the Internet grew larger, and the world grew smaller. Many battles were fought in the virtual „dungeons“ (and many social-lives were lost). Many people made many advances in the way MUDs worked, and the way they could be used. There was born the DikuMUD, the TinyMUD, the LPMUD, MUSEs, MUSHes, MUCKs, MOOs, and many others. The worlds became more elaborate, and it became possible to program sophisticated actions, and reactions, and commands, and entire system designs, all from within the virtual worlds themselves, changing the reality itself as one was inside it.

Today, MUD-type systems are used for social communities, scientific forums, educational environments, process control systems, business conferencing systems, and nearly anything else people have come up with. Many MUD servers have been adapted to work with graphical web interfaces, and even 3D VR perspectives. Artificially intelligent „softbots“ (software robots), both from outside and programmed directly into MUDs interact with users and with the environments, while users can build and shape their environments, and most importantly, share information, ideas, resources, and communities online, from thousands or millions of miles away.

First steps in a MOO

In most MUDs, characters are anonymous. People who become friends can exchange real names and email addresses, but many choose not to. Conventions about when it is acceptable to talk about „real life“ vary between communities. In most MUDs, people begin to talk more about real life when they get to know someone better.

From the software point of view, a MUD is a program that accepts „connections“ from multiple users across some kind of network (e.g., telephone lines or the Internet) and provides to each user access to a shared database of „rooms“, „exits“, and other objects. Each user browses and manipulates this database from „inside“ one of those rooms, seeing only those objects that are in the same room and moving from room to room mostly via the exits that connect them. A MUD, therefore, is a kind of virtual reality, an electronically-represented „place“ that users can visit.

Assignment 1. Create your own MOO character.

The typical MUD user interface is most reminiscent of old computer games like Adventure and Zork; a typical interaction is shown in Figure. Three major factors distinguish a MUD from an Adventure-style computer game, though:

1. A MUD is not goal-oriented; it has no beginning or end, no „score“, and no notion of „winning“ or „success“. In short, even though users of MUDs are commonly called players, a MUD isn't really a game at all.
2. A MUD is extensible from within; a user can add new objects to the database such as rooms, exits, 'things', and notes. Certain MUDs even support an embedded programming language in which a user can describe whole new kinds of behavior for the objects they create.
3. A MUD generally has more than one user connected at a time. All of the connected users are browsing and manipulating the same database and can encounter the new objects created by others. The multiple users on a MUD can communicate with each other in real time.

This last and very important factor has a profound effect on the ways in which users interact with the system, it transforms the activity from a solitary one into a social one.

```
>look
Corridor
The corridor from the west continues to the east here,
but the way is blocked by a purple-velvet rope
stretched across the hall. There are doorways leading
to the north and south.
You see a sign hanging from the middle of the rope here.
>read sign
This point marks the end of the currently-occupied
portion of the house. Guests proceed beyond this point
at their own risk.
-- The residents
>go east
You step disdainfully over the velvet rope and enter
the dusty darkness of the unused portion of the house.
```

Figure 8 A typical MUD database interaction

When a person first logs onto a MUD, he or she creates a character. The person selects the character's name and gender, and writes a description of what the character looks like. It is possible for a character to be male or female, regardless of the gender of the player. In many MUDs, a character can also be neuter or even plural.

MOO basic commands are given in Appendix 3.

MUDs are organized around the metaphor of physical space. You can „talk“ to anyone in the same virtual room. When you connect to MediaMOO (MUD at the Media Lab), you see the description:

```
>connect guest
Okay,... guest is in use. Logging you in as
'Green_Guest'
*** Connected ***
The LEGO Closet
It's dark in here, and there are little crunchy
plastic things under your feet! Groping around,
you discover what feels like a doorknob on one wall.
Obvious exits: out to The LEGO/Logo Lab
```

The core of MediaMOO is a virtual representation of the MIT Media Lab. Typing „out“ gets you to the „LEGO/Logo Lab,“ a central work area for the lab's Epistemology and Learning (E&L) research group:

```
>out
The LEGO/Logo Lab
The LEGO/Logo Lab is a happy jumble of little and
big computers, papers, coffee cups, and stray pieces of LEGO.
Obvious exits: hallway to E&L Hallway, closet to
The LEGO Closet, and sts to STS[3] Centre Lounge
You see a newspaper, a Warhol print, a Sun
SPARCstation IPC, Projects Chalkboard, and Research Directory here.
Amy is here.
```

```
>say hi
You say, „hi"
Amy says, „Hi Green_Guest! Welcome!"
```

```
The Thin Blue Line arrives and slows to a stop.
The conductor of The Thin Blue Line cries, „Next stop is Ballroom Foyer."
The conductor of The Thin Blue Line cries, „All aboard!"
The Thin Blue Line moves out slowly, gathering speed as it vanishes into the distance.
```

In this transcript, a guest connects and speaks with a real person (Amy). Each person could be anywhere in the world with an Internet connection. As Green_Guest and Amy were talking, a train came through. If Green_Guest were to type „enter train,“ it would give him/her a tour of interesting places around MediaMOO. The train system was programmed by a MediaMOO user named „Moose“ who is a graduate student at Brown University.

Most inter-player communication on MUDs follows rules that fit within the framework of the virtual reality. If a player 'says' something (using the say command), then every other player in the same room will 'hear' them. For example, suppose that a player named Vira typed the command

say Can anyone hear me?
Then Vira would see the feedback
You say, „Can anyone hear me?“
and every other player in the same room would see
Vira says, „Can anyone hear me?“

Similarly, the emote command allows players to express various forms of 'non- verbal' communication. If Vira types
emote smiles.
then every player in the same room sees
Vira smiles.

It sometimes happens that one player wishes to speak to another player in the same room, but without anyone else in the room being aware of the communication. If Vira uses the whisper command

whisper „I wish he'd just go away...“ to Ceco
then only Ceco will see
Vira whispers, „I wish he'd just go away...“
The other players in the room see nothing of this at all.

Finally, if one player wishes to say something to another who is connected to the MUD but currently in a different and perhaps 'remote' room, the „page“ command is appropriate. It is invoked with a syntax very like that of the whisper command and the recipient sees output like this:

You sense that Vira is looking for you in The Hall.
He pages, „Let's go out for a beer!“

Every object in a MUD optionally has a textual description which players can view with the look command. For example, the description of a room is automatically shown to a player when they enter that room and can be seen again just by typing 'look'. To see another player's description, one might type 'look Bombone'.

Players can set or change their descriptions at any time. The lengths of player descriptions typically vary from short one-liners to dozen-line paragraphs.

MUD players typically spend their connected time socializing with each other, exploring the various rooms and other objects in the database, and adding new such objects of their own design. They vary widely in the amount of time they spend connected on each visit, ranging from only a minute to several hours; some players stay connected (and almost always idle) for days at a time, only occasionally actively participating.

MOOs can be divided into three main groups:

1. educational, Foreign Language, EFL/ESL (English as a Foreign Language/English as a Second Language), and research MOOs;
2. gaming MOOs;
3. social MOOs.

These categories are in no way mutually exclusive. Often social MOOs, and sometimes educational and research MOOs, are set in a science-fiction (sci-fi) or fantasy setting, much like a gaming MOO. All MOOs have social aspects. With the exception of ZenMOO, all MOOs are programmable, and could thus be considered research MOOs.

MUDs are increasingly being used for more „serious“ purposes. David Van Buren of the California Institute of Technology and Pavel Curtis of Xerox PARC has developed a MUD to enhance professional community among astrophysicists called AstroVR (Curtis and Nichols, 1993). The

MediaMOO project is designed to enhance professional community among media researchers (Bruckman and Resnick, 1993). As of February 1994, MediaMOO had over 900 participants from twenty-three countries.

In conclusion, here are some brief remarks on MUDs (MOOs) (Schneider, 1998):

- many persons can connect simultaneously to a server.
- the MUD has a spatial organization, e.g. people interact with people or objects primarily within „rooms“.
- within a MOO, many real time communication actions exist, like „saying“ or „emoting“ things publically to the persons in a „room“, paging people elsewhere, „whispering“ messages, etc.
- asynchronous communication tools include internal e-mail, Newsgroups, News papers, tutorial rooms, „notice boards“, etc.
- MOOs are extensible, and most imaginable (text-based) objects and features can be programmed.
- MOOs also can be used as backend for various network services. A good example are http servers.

Assignment 2. Solve the Mona-Lisa mystery game

Why multi-user worlds for education?

(Schneider, 1998) presents some arguments in favour of VEEs. Here they are:

- collaboration is good for learning (and such environments are collaborative);
- using VEEs will lead to the discovery and use of new ways of teaching;
- they are easily to be used in distance teaching,
- virtual environments foster a sense of community that in many cases favors learning (users develop an „electronic identity“ and form on-line communities)
- „interactivity is a key to learning“ ;
- experience is actively constructed and reconstructed through direct interaction with the world
- knowledge is experience in a real learning experiment (immersion) and VEEs are immersive (either sensory or social, sometimes both); Immersion in an experience is also often as enhancement factor for learning (experience is actively constructed and reconstructed through direct interaction with the world). As states (Ackermann, 1994a, p. 13), „interactivity is a key to learning“ and „An increasing number of software designers, cognitive scientists and educators have come to the view that experience is actively constructed and reconstructed through direct interaction with the world, and that, indeed, knowledge is experience“.
- social interactivity is a key to learning (apprenticeship)
- multi-user worlds have potential for being „cyberspace integrators“.

Papert (in Mindstorms) describes the Brazilian „Samba Schools“ which are not really schools, but social clubs in which people of all ages and levels of experience work together in both formal and informal ways. He is inspired by the different relationship to learning the members develop, and the way in which learning becomes a community process. Traditional schools generally undermine the development of such a learning culture by excessively formalizing learning and segregating people by age and experience.

In Samba schools learning is (quoted from Bruckman's MOOSE Crossing Ph.D. proposal):

- self-motivated,
- richly connected to popular culture,
- focused on personally-meaningful projects,
- community based,

- an activity for people of all ages to engage in together,
- life long experts as well as novices see themselves as learners, and
- situated in a supportive community.

Schneider (1998) claims that „Text-based virtual worlds on the Internet (MUDs) can have each of these features“, but

1. those environments are not really user friendly
2. „Embedded“ learning domains are limited so far, mostly:
 - programming (interactive objects, other internet applications)
 - verbal skills (how to articulate, describe, organize)
 - social skills (how to deal with people)

Schneider and Godard (1996) give the following reasons for using VEs for education:

1. VEs might be productive for several reasons:

- VEs are collaboration tools. Many studies have shown that collaborative learning and collaborative work is efficient, some conditions for either success or failure are well known, others are still unknown (Dillenbourg and Schneider, 1995).
- Immersion in an experience is also often as enhancement factor for learning
- A large segment of researchers view VEs as a change to renew pedagogies (in the MUD world we can cite [Fanderclai, 1995] and [Moshell et al., 1995]). Transposing the traditional classroom into cyberspace (e.g. [Speh, 1994] for the MOO world) is interesting in many cases (e.g. for Distance Teaching).

2. VEs should be „cyberspace desk tops“ with multiple communication and media layers.

3. VEs must adapt to the users and not the other way round.

4. VEs need objects that can be manipulated:

- VEs can be seen as a social context for propagating constructionism (e.g. students manipulating „symbolic physical“ objects ([Moshell et al., 1995]) or „real virtual physical objects“).
- VEs can be enhanced with artificial agents or more generally can integrate Human-Computer Collaborative Learning Systems (HCCLS) with Computer-Supported Collaborative Learning Systems (CSCLS).

5. Navigation Interfaces Navigation interfaces are a crucial issue (see [Dieberger et Tromp, 1993], [Dieberger, 1994], [Girardin, 1995]).

6. „Real“ and „social VR“ are complementary.

„Real VR“ presenting the user with the illusion of being in a 3-D world of computer generated objects (e.g. using goggles, gloves) and „social VR“ (e.g. MUDs) are „worlds a part“.

The question of „what kind of VE do we really need for various aspects of education and collaborative work“? Could there be an integrating standard?“ it is still an open question that has to be answered in the near future.

4.4 Computer Mediated Communication

Definitions

As used in Computer-Mediated Communication and the Online Classroom (Hampton Press, 1995), the term computer-mediated communication (CMC) signifies the ways in which telecommunication technologies have merged with computers and computer networks to give us new tools to support teaching and learning. CMC describes the ways we humans use computer systems and networks to transfer, store, and retrieve information, but our emphasis is always on communication. In our model, the computer network is primarily a mediator for communication rather than a processor of information. As it is currently used to support instructional purposes, CMC provides electronic mail and real-time chat capabilities, delivers instruction, and facilitates student-to-student and student-to-teacher interactions across a desk or across the world. These uses are enabling and promoting several paradigmatic shifts in teaching and learning, including the shift from instructor-centered distance education to student-centered distance learning and the merging of informal dialogues, invisible colleges, oral presentations, and scholarly publications into a kind of dialogic (or even multilogic) virtual university.

Educators often categorize the use of instructional CMC in three ways:

1. conferencing,
2. informatics, and
3. computer-assisted instruction (CAI).

Computer conferencing provides e-mail, interactive messaging, and small and large group discussion. Informatics (repositories or maintainers of organized information) include library online public access catalogs (OPACs), interactive access to remote databases, program/data archive sites (e.g., archives of files for pictures, sound, text, movies), campus-wide information systems (CWIS), wide-area information systems (WAIS), and information managers, such as Gopher and Veronica. In CAI, the computer is used to structure and manage both the presentation of information and the possible responses available to the human user.

Uses of computer conferencing, informatics, and CAI include:

- mentoring, such as advising and guiding students
- project-based instruction, either within the classroom or in projects involving community, national, or international problem solving
- guest lecturing, which promotes interaction between students and persons in the larger community
- didactic teaching, that is, supplying course content, posting assignments, or other information germane to course work
- retrieval of information from online information archives, such as OPACs, ERIC, and commercial databases
- course management, for example, advising, delivery of course content, evaluation, collecting and returning assignments
- public conferencing, such as discussion lists using main frame Listserv software
- interactive chat, used to brainstorm with teachers or peers and to maintain social relationships
- personal networking and professional growth and such activities as finding persons with similar interests on scholarly discussion lists
- facilitating collaboration
- individual and group presentations

- peer review of writing, or projects involving peer learning, groups/peer tutorial sessions, and peer counseling
- practice and experience using emerging technologies that may be intrinsically useful in today's society
- computer-based instruction, such as tutorials, simulations, and drills.

Here is a short attempt list various criteria for classifying CMC media (WWW6 workshop, 1997).

- Co-presence: Are participants „present“ in the same location ?
- Visibility: Are participants visible ?
- Audibility: Can people talk and listen ?
- co-temporality: Do people communicate at the same time ?
- sequencibility: How fast do we get a reaction usually ?
- reviewability: Can people review what has been uttered ?
- revisability: Can people revise before they send something ?
- archivability: Can the result be stored easily ?
- user numbers: How many users can participate at the same time ? ?
- proxemics: At what distance can people be within a medium
- simultaneity: Can people „utter“ at the very same time ? turn-taking ?
- multi-channels: How many channels can participants use ?
- multi-media: What kind of media does it support ?
- intrusiveness: How intrusive is the tool ?
- structuredness: How does the tool pre-structure conversation ?

The properties of computer-mediated communication that make it at once very much like face-to-face communication and also very different from it make MOOs interesting environments for a wide range of research. Sociologists like Sherry Turkle study CMC to learn how people define their identities and their social relationships in cyberspace. People studying computer-supported cooperative work look at MOOs to see how they might be used in commercial settings to facilitate collaboration between co-workers. Linguists study the communication between players in a MOO to learn how people alter their language use in the MOO's artificial setting, hoping to get a better understanding of language use in broader contexts. Researchers in artificial intelligence and computational linguistics look to see how software agents might be built to interact with people in text-based media in CMC can ways that are natural and useful.

Computer supported collaborative learning (CSCL)

Tele-learning designates new forms of distance or of computer mediated learning, where the distance is not only distance in space or time as in traditional distance learning, but the mediation of learning activities served by information and communication technologies such as multimedia shared workspaces, multimedia communication (e.g., chat boxes), or multimedia servers.

Collaborative telelearning emphasises the collaborative interaction between students in a tele-learning environment. Collaborative learning is a collection of perspectives based on principles of interpersonal interaction (Sorensen, 1997).

There are three perspectives which place emphasis on different goals (Fjuk, 1998):

1. Joint construction of knowledge (e.g., joint problem-solving by mutual refinement);
2. Joint negotiation of alternatives (e.g., through argumentation);
3. Students rely on each other (and teacher) as a resource to support their own learning and to get feedback

Computer supported collaborative learning (CSCL) is an emerging paradigm in its formative stages (Koschmann, 1996) that focuses on the use of technology as a mediational tool within collaborative methods of instruction (e.g., peer learning and tutoring, reciprocal teaching, project- or problem-based learning, simulations, games). It is an approach to education that emphasises an understanding of language, culture and other aspects of the social setting (Scott, Cole & Engel, 1992)

The intellectual heritage of CSCL is based on three foundations:

1. cultural-historical psychology (e.g., Vygotsky, 1978; Leontiev 1975; Davydov, 1988) that stresses the cultural basis of human intellect (e.g., learning first inter-individual then intra-individual);
2. social constructivism (e.g., Doise, 1990) that focuses on the individual's development with respect to social interaction (extension of Piaget's theory) and
3. situated cognition (e.g., Suchmann, 1987; Lave, 1988) which stresses the learning environment within which learning takes place (includes both physical and social contexts); learning is entering a „community of practice“ (e.g., Lave, 1988, Brown, Collins & Duguid, 1989).

We have to note here that collaboration = communication + coordination and that cognitive development happens through interactions between students or between students and rich, knowledgeable environments.

Whilst primarily a framework for providing an explanatory account of distributed cognition, the distributed cognition approach has great potential for being applied to CSCL. Rogers and Halverson (1995, 1996) have given tutorials with this endeavour in mind at ECSCW'95 (Stockholm) and CSCW'96 (Boston). At one level, it can be very useful in identifying problems with existing work practices and use of technological artefacts. At another, it can highlight what is salient and important in existing system designs, that needs to be retained in subsequent designs intended to augment, enhance or replace current systems and ways of working.

Following are some of research findings of computer supported collaborative learning (CSCL):

- A numerous research evidence suggests that a combination of group rewards and strategy training produces much better outcomes than either one alone (Fantuzzo et al., 1992).
- the results of ACOT's two years (1986-87) study of seven classrooms that represented a cross section of America's K-12 schools are promising. Teachers are able to translate traditional text-based instructional approaches to the new electronic medium. Student deportment and attendance improved across all sites, their attitude towards self and learning showed improvement as well. In terms of test scores, at the very least, students are doing as well as they might without all of the technology and some are clearly performing better (Apple Research Labs Publications).
- Sherry and Myers (1996) study group dynamics of graduate students collaboratively design WWW process. They confirm Scardamalia et al. (1994) „World 3“ view that the group becomes a self-reflective, and self-organizing system that each member contributes her own expertise and, in turn, learning new skills and extending the group knowledge based.
- Study shows that the more skilled teacher participates with the technology, the more positive attitudes they have developed toward technology (Zhao & Compbell, 1995).
- There is substantial evidence that students working in groups can master science and mathematics materials better than students working alone (Slavin, 1989).
- King (1989) observes verbal interaction and problem solving behaviour of small collaborative peer groups working on CAI tasks. He finds successful group involved in more task talks than

social talks. They ask more task related questions, spend more time on strategies use, and obtain higher elaboration scores than did unsuccessful groups.

- Weir (1992) indicates that both teachers and researchers find that students who work together on „real world problems show increased motivation, deeper understanding of the concept and an increased willingness to tackle difficult questions that they cannot answer alone."
- This focus on authenticity and experiential learning is reiterated in numerous articles.
- A series of CSILE studies conducted by Scardamalia and Breiter, indicate that students gain deeper understanding and collaboratively construct knowledge while working in CSILE environments.
- CSCL environment can accommodate a larger group size (can up to 20, studies show size =60 is too big) that increases idea generation and decision making. The ideal size of face-to-face group is four.
- The role of the teacher will shift from primary source of knowledge to that of expertise in learning. A good teacher should be an expert learner, who can facilitate students' learning and information searching (Riel, 1994).

4.5 Intelligent Agents for Education

First definitions

Nowadays, every computer scientist and not only, speaks about agents. To start with a joke let us see the first definition of an agent: „An agent is a software thing that knows how to do things that you could probably do yourself if you had the time.“ (Ted Selker, IBM Almaden Research Centre)

But the field is extremely serious. Here is a reason:

„Agents are here to stay, not least because of their diversity, their wide range of applicability and the broad spectrum of companies investing in them. As we move further and further into the information age, any information-based organisation which does not invest in agent technology may be committing commercial hara-kiri.“ (Hyacinth S. Nwana, renowned expert in intelligent agents).

Why agents are attracting attention

Agents are being explored as a means to many and varied pressing ends in today's information and manufacturing economy (Agents SourceBook):

- To put intelligence into user interfaces to enable unskilled users to get more out of computing applications.
- To personalize applications and services to meet users' preferences, goals, and desires.
- To manage the retrieval, dissemination, and filtering of the vast amounts of information available on enterprise networks, value-added networks, and especially the Internet.
- To enable electronic commerce in various forms.
- To manage flexible manufacturing cells (robotics)
- To support teaching and learning (a reason we have added).

Agent based computing has been described as „the next significant breakthrough in software development“. The UK based consultancy firm Ovum has predicted that agent technology industry would grow from a US \$37 million in 1994 to US \$2.6 billion worldwide by the year 2000.

Definitions of what is an agent

Currently there is no widely accepted universal definition of a software agent. A dictionary definition of the term „agent“ follows: „A person or thing that acts or is capable of acting or is empowered to act, for another.“ A dictionary definition for the term „software agent“ is: „A computing entity (piece of software) that performs user delegated tasks autonomously.“

Other worth mentioning definition say:

„An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through effectors.“ (from [Russell and Norvig, 1995, p. 33] (see also Fig. 7). This is the AIMA Agent (AIMA is an acronym for „Artificial Intelligence: a Modern Approach“, a remarkably successful new AI text that was used in 200 colleges and universities in 1995 [Russell and Norvig, 1995]). The authors were interested in software agents embodying AI techniques. Clearly, the AIMA definition depends heavily on what we mean by „environment“, and on what sensing and acting mean.

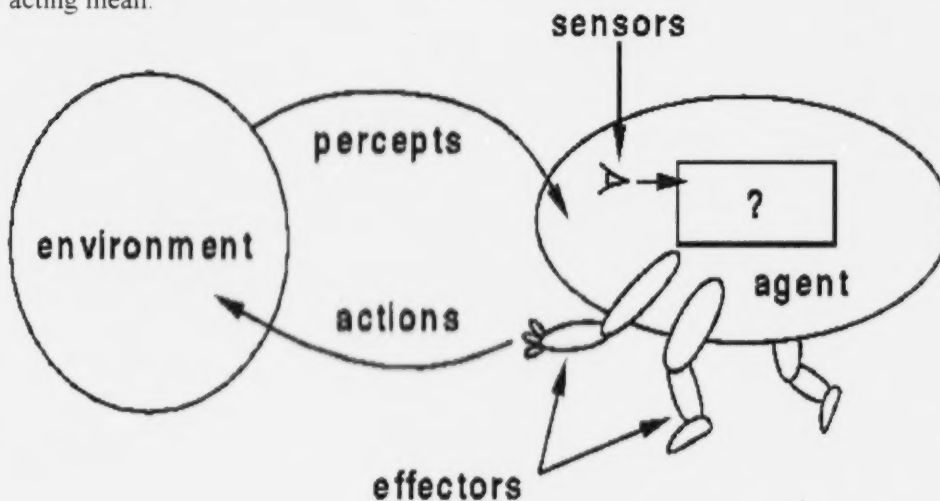


Figure 9 A definition for an agent

The problem with this definition is that if we define the environment as whatever provides input and receives output, take input to be sensing, and producing output to be acting then every program is an agent. So, by restricting the notions of environment, sensing and acting we might arrive at a useful distinction between agent and program.

A weak notion (from [Woodridge and Jennings, 1995, p. 2]) says that essential properties of agents are:

- autonomy: agents operate without direct intervention of humans, and have control over their actions and internal state;
- social ability: agents interact with other agents (and possibly humans) via an agent communication language;
- reactivity: agents perceive their environment and respond in a timely and rational fashion to changes that occur in it;
- pro-activeness: agents do not simply act in response to their environment, they are capable of taking the initiative (generate their own goals and act to achieve them).

A stronger notion (ibidem) says:

An agent has mental properties, such as knowledge, belief, intention, obligation. In addition, and agent has other properties such as:

- mobility: agents can move around from one machine to another and across different system architectures and platforms;
- veracity: agents do not knowingly communicate false information;
- benevolence: agents always try to do what they are asked of;
- rationality: agents will try to achieve their goals and not act in such a way to prevent their goals from being achieved.

Other possible definitions are:

The MuBot Agent (<http://www.crystaliz.com/papers/mubot.htm>) „The term agent is used to represent two orthogonal concepts. The first is the agent's ability for autonomous execution. The second is the agent's ability to perform domain oriented reasoning.“ This pointer at definitions come from an online white paper by Sankar Virdhagriswaran of Crystaliz, Inc., defining mobile agent technology. Here we note the fact that autonomous execution is clearly central to agency.

The Maes Agent (Maes, 1995, page 108): „Autonomous agents are computational systems that inhabit some complex dynamic environment, sense and act autonomously in this environment, and by doing so realize a set of goals or tasks for which they are designed.“ Pattie Maes, of MIT's Media Lab, is one of the pioneers of agent research. She adds a crucial element to her definition of an agent: agents must act autonomously so as to „realise a set of goals.“ Also environments are restricted to being complex and dynamic.

The KidSim Agent (Smith, Cypher and Spohrer, 1994) „Let us define an agent as a persistent software entity dedicated to a specific purpose. 'Persistent' distinguishes agents from subroutines; agents have their own ideas about how to accomplish tasks, their own agendas. 'Special purpose' distinguishes them from entire multifunction applications; agents are typically much smaller.“ The explicit requirement of persistence is a new and important addition here.

The Hayes-Roth Agent (Hayes-Roth, 1995) „Intelligent agents continuously perform three functions: perception of dynamic conditions in the environment; action to affect conditions in the environment; and reasoning to interpret perceptions, solve problems, draw inferences, and determine actions.“

The IBM Agent (<http://activist.gpl.ibm.com:81/WhitePaper/ptc2.htm>) „Intelligent agents are software entities that carry out some set of operations on behalf of a user or another program with some degree of independence or autonomy, and in so doing, employ some knowledge or representation of the user's goals or desires.“ This definition, from IBM's Intelligent Agent Strategy white paper, views an intelligent agent as acting for another, with authority granted by the other, as our first dictionary definition has said, too.

The SodaBot Agent [Michael Coen, <http://www.ai.mit.edu/people/sodabot/slideshow/total/P001.html>] „Software agents are programs that engage in dialogs [and] negotiate and coordinate transfer of information.“

The Foner Agent [<http://foner.www.media.mit.edu/people/foner/Julia/>]. Foner requires much more of an agent. His agents collaborate with their users to improve the accomplishment of the users' tasks, i.e. the agent is collaborative.

The Brustoloni Agent (Brustoloni, 1991; Franklin, 1995, p. 265) „Autonomous agents are systems capable of autonomous, purposeful action in the real world.“ The Brustoloni agent, unlike the prior

agents, must live and act „in the real world.“ This definition excludes software agents and programs in general and might include intelligent robotic agents for example.

An end user taxonomy of agents is given in Appendix 4.

Agencies

Software agents, like people, can be most useful when they work with other software agents in performing a task. A collection of software agents that communicate and cooperate with each other is called an agency. Agents are the entities that actually perform the task that has been assigned to them. The agency is the network wide infrastructure that co-ordinates the work of separate agents. It is the existence of the Agency that enables the ‘cooperative society’ of agents.

There is a minimum set of common features that typify a software agent.

A software agent is autonomous; the agent is capable of operating as a standalone process and performing actions without user intervention.

A software agent is communicative; it communicates with the user, other software agents, or other software processes.

A software agent is perceptive; it is able to perceive and respond to changes in its environment.

Software agents, like people, can possess different levels of competence at performing a particular task. While software agents must be autonomous, communicative, and perceptive, they can have different levels of competence (this notion is more suited than that of intelligence we believe) as determined by their programs.

Other characteristics or attributes of a software agent are:

Delegation: The agent performs a set of tasks on behalf of a user (or another agent) that are explicitly approved by the user.

Communication skills: The agent needs to be able to interact with the user (and sometimes other agents) to receive task delegation instructions, and inform task status and completion through an agent-user interface or through an agent communication language.

Autonomy: The agent operates without direct intervention (e.g., in the background) to the extent of the user’s specified delegation. The autonomy attribute of an agent can range from being able to initiate a nightly backup to negotiating the best price of a product for the user.

Monitoring: The agent needs to be able to monitor its environment in order to be able to perform tasks autonomously.

Actuation: The agent needs to be able to affect its environment via an actuation mechanism for autonomous operation.

Intelligence (competence): The agent needs to be able to interpret the monitored events to make appropriate actuation decisions for autonomous operation.

In addition, some agents may have other attributes, among them:

Temporal continuity: Many agents need to be continuously running processes, not ‘one-shot’ computations that map a single input to a single output, then terminate.

Character: Some agents have a well-defined, believable ‘personality’ and emotional state.

Adaptive: Some agents automatically customise themselves to the preferences of their users based on previous experience. These agents also automatically adapt to changes in their environment.

Mobile: Some agents need to be able to transport themselves from one machine to another and across different system architectures and platforms.

Multiagent systems research, a subfield of artificial intelligence, studies the interactions of computational agents.

Architectures for agents

The following simple architectures are taken from (Russell and Norvig, 1995, Chapter 2):

Simple reflex agent

The agent works by finding a rule whose condition matches the current situation, as defined by perception, and then doing the action associated with the rule. The agent has no memory.

A reflex agent with internal state

The agent works by finding a rule whose condition matches the current situation, as defined by perception and by the stored internal state, and then doing the action associated with the rule. The internal state acts as a memory and allows for a better selection of the rule to apply.

An agent with explicit goals

The agent has an explicit goal, and when choosing an action it will select an action that achieves the goal. The decision process requires some planning ("what will happen if I do action A"). This agent is more flexible than the simple reflex agent, and is capable of achieving goals.

A utility-based agent

The agent has a utility function that maps a state onto real number that describes how well the agent is performing. This agent not only achieves goals, but it maximizes some measure of performance.

Application Fields

The concept of software agent has been used in heterogeneous research and application fields: Computer-Supported Cooperative Work (with a focus on co-operation), Human-Computer Interaction (with a focus on adaptivity), and Artificial Intelligence (with a focus on distributed problem solving) are three areas which put efforts in the development of software agents. Moreover, agents may play an important role during a design process, in workflow and network management, in messaging, in information retrieval, and in mobility management (Guilfoyle & Warner 1994).

The next table from Chapter 1 of the „Agent Sourcebook“ gives a thumbnail view of the broad functional benefits of agent technology:

Table 2 Benefits of Agent Technology

Feature	Advantage	Benefit.
Automation	Perform repetitive tasks	Increased productivity.
Customization	Customize information interaction	Reduced overload
Notification	Notify user of events of significance	Reduced workload
Learning	Learn user(s) behavior	Proactive assistance
Tutoring	Coach user in context	Reduced training
Messaging	Perform tasks remotely	Off-line work

Software agents for education and learning

In the AIED community agents refer to tutors, coaches, co-learners. In the WWW and Internet community they mostly name some kind of smart search engines. I would like to review here some types of software agents that promise to be of high interest in designing high performance learning systems.

1. Chatterbots

A chatterbot is a „bot“ that attempts to simulate the conversation or „chat“ of a human being. The word „bot“ is derived from „robot“, and refers to a computer program that gathers information or performs a service, usually on some regular schedule. A bot (sometimes called also an agent) typically searches the Internet, gathers information relevant to your interests, and presents it to you on a daily or other periodic basis.

Bots gained notoriety on IRC (Internet chat groups) where they served such functions as greeting new participants, monitoring the use of offensive language, and sometimes being downright annoying. Today they retrieve information, and respond to events on Usenet, the Web and other corners of cyberspace. While some bots silently roam the net at your bidding, others are interactive, such as chatterbots - the kind of bot you can talk to.

Chatterbots such as „Eliza“ and „Parry“ were well-known early attempts at creating programs that could at least temporarily fool a real human being into thinking they were talking to another person. A chatterbot can be thought of as the spokesperson for „Artificial Intelligence“ (AI). Chatterbots have been used in the online interactive games called Tinymuds. A single player may be able to interact with a chatterbot while awaiting other „live“ players.

You might find a chatterbot in a future educational system that will assist you in learning or doing a specific task. You might not realize that it is not a human agent that you are talking with.

Here are some notorious bots.

1. Eliza

In 1966, Massachusetts Institute of Technology Professor Joseph Weizenbaum wrote a computer program for the study of natural language communication between man and machine. She was initially programmed in a mere 240 lines of code, and simulates a psychotherapist by answering questions with questions. Here's a sample conversation with Eliza:

Me: Hi there. I've got a few questions to ask...

Eliza: Please go on.

Me: Well, I think I have a drinking problem.

Eliza: But you are not sure you have a drinking problem?

Me: I'm sure. All my shirts are stained in front.

Eliza: Why do you say your shirts are stained in front?

Me: Because that's where I dribble my tomato juice.

Eliza: Do any other reasons not come to mind?

Although Eliza is almost 32 years old, you can still try it out (<http://www-ai.ijs.si/eliza/eliza.html>).

Eliza's operating system is quite simple. She analyzes each question that you ask her looking for keywords or multiple words that she recognizes. She has been given programmed responses to these keywords and sentence structures.

2. *Shallow Red Bot*

Launched online in June, 1997 by Neuromedia, Inc. of San Francisco, Shallow Red works in the sales and marketing field, using chat, web pages, and Samba music to answer basic questions about Neuromedia's products and how bots are created and used. Most importantly, Shallow Red is a script running on an underlying bot engine. This design separates content (text and HTML) of the bot from the complex AI underneath.

3. *ALICE*

ALICE (Artificial Linguistic Internet Computer Entity) is a more sophisticated program that creates the illusion of natural conversation. ALICE can even tell lies, spread gossip, and is credited with influencing several real world happenings.

Assignment 3. Talk with two chatterbots and compare them.

2. Observer Agents

The notion of agent has been present more or less explicitly in Intelligent Learning Environments for some time. Similar types of agents are now investigated in Internet-based systems.

Sub-agents. These autonomous software entities which „carry out tasks for the user“ are deemed by many to be the next revolution in computing. Since the World Wide Web (WWW) provides an evolving environment of voluminous amounts of unorganized information. Information sources appear and disappear, making information filtering a crucial capability of any Internet-based information gathering system. To date most research on Intelligent Information Agents has dealt with a user interacting with a single agent that has general knowledge and is capable of performing a variety of user delegated information finding tasks (e.g., Etzioni & Weld, 1994; Lieberman, 1995). Other examples of tasks accomplished by such agents are e-mail filtering (Lashkari & al., 1995) room reservation and calendar managing (Bocionek, 1995) or finding people sharing same interests (Foner, 1996). Many of these systems use machine learning techniques to non-intrusively infer the user preferences. Distributed multi-agent systems have been used to overcome limits of machine-learning methods such as:

- (1) a single general agent need an enormous amount of knowledge to deal effectively with user information requests that cover a variety of tasks,
- (2) a centralized information agent constitutes a processing bottleneck, the required processing would overwhelm a single agent which would constitute a „single point of failure“,
- (3) a single agent needs considerable reprogramming to deal with the appearance of new agents and information sources in the environment.

Co-agents. In symmetrical systems (Dillenbourg and Baker, 1996), human and artificial agents can perform the same actions. The idea of co-learner, originally introduced as an alternative to learner modelling (Self, 1986), was re-used within various learning paradigms: collaborative learning (Chan & Baskin, 1988; Dillenbourg and Self, 1992), competitive activities, reciprocal tutoring] (Chan, 1996), learning by teaching and teacher training (Ur & van Lehn, 1995).

Super-agents. Most intelligent learning environments include agents (coach, tutor, expert,...) which provide solutions and monitor the actions of users. In the case of multi-user learning environments, super-agents have to monitor the interactions among users. They can be teachers who analyze interactions to detect when they should intervene, judges or referees who intervene only in case of conflict, etc. For instance, the Belvedere system (Suthers & al., 1995) is dedicated to support critical discussions of science issues. Students build a diagrammatic representation with arguments. They can invoke an advisor who points to specific parts of the diagram and proposes ways of extending or revising it. Some argument structures are expected to be more effective in stimulating critical

discussions among students. In COSOFT (Hoppe, 1995), a super-agent compares its model of on-line students and invites students with specific lacks to collaborate with students who possess the missing skills. Critic systems can also be viewed as super-agents, since they evaluate the user's work, but without a pedagogical perspective: „Critics do not necessarily solve problems for users. The core task of critics is the recognition and communication of deficiencies in a product to the user.“ (Fisher et al., 1991).

Observers. They are agents who collect information regarding users interactions, aggregate observations into high level indicators and display these indicators to the human coach or to the users themselves. One cannot a priori set up collaborative settings which guarantee effective collaboration, hence a coach must monitor the interactions (Dillenbourg et al, 1995). Observers would support this monitoring process both for human and artificial coaches. These indicators could also be shown to the subjects. It is interesting to study how a pair could use them to regulate its interactions and a major question is whether the socio-cognitive behavior of the peer changes when it gets information about its interaction. First experiments with observer agents were reported in (Buiu, 1997).

An example of observer agents that are implemented in a virtual reality world (MOO) using the Tcl/Tk scripting language and the MOO programming language is given in Appendix 5.

3. Collaborative Agents

Roschelle and Teasley (1995) defined collaboration as a „Coordinated, synchronous activity that is the result of a continued attempt to construct and maintain a shared conception of a problem“. More precisely, the 'Joint Problem Space' (JPS) „[...] is a shared knowledge structure that supports problem solving activity by integrating (a) goals (b) descriptions of the current problem state, (c) awareness of available problem solving actions, and (d) associations that relate goals, features of the current problem state, and available actions“.

It is important to make a distinction between cooperation and collaboration. To perform its own goal or task, an agent might need to cooperate with others (artificial or human agents) because it might be unable to do it itself, or others could do it more efficiently. Specific to co-operation is that the task to be solved is divided into sub-tasks distributed to different agents.

Speaking about collaboration, we have to notice that some goals may concern all agents, may be realized individually or by several agents. Collaboration might be defined as a mutual engagement of participants (artificial and human agents) in a coordinated effort to solve the problem together. Some global goals might be only achieved by several agents collectively. Here it appears the problem of splitting the global task and distributing it to agents (DAI systems).

Implicit coordination mechanisms do often occur in collaborative problem solving systems. In real life tasks, coordination is rarely done explicitly and much more often it is done implicitly, i.e. one agent infers from indirect clues that the other agent(s) has collected some information. As a result, it will spontaneously search for other information. Implicit coordination obviously reduces the communication costs.

Collaborative software agents (agents able to collaborate with a human agent, or „cobots“ (collaborative software robot), a term we introduce here from the first time) could be of high interest in distance education or in on-line assistance. The study of human-human collaboration might lead to interesting implication on designing artificial collaborative agents. In (Dillenbourg, Traum, Schneider, Jermann and Buiu, 1997) there are reported experiments on collaborative problem solving in a virtual environment. The authors' long term goal is to develop computational models for supporting human-computer collaboration (Dillenbourg, 1996). The specific goal was to study how

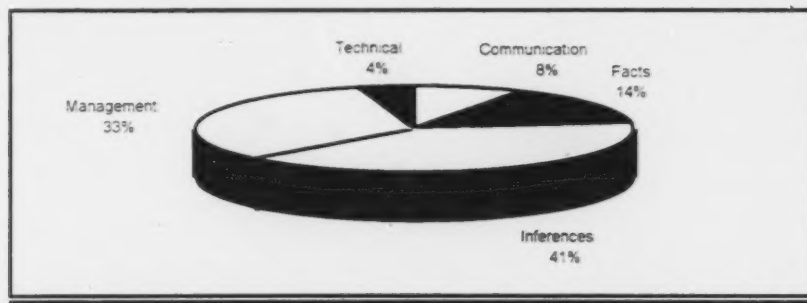
two human subjects elaborate a shared solution of a problem and how the drawings they make on a shared whiteboard help to support mutual understanding.

The test problem they have used is the mystery game described in Assignment 2. They ran the mystery solving experiments with 20 pairs of human agents. The average time for solving it was two hours. The experiments have allowed the authors to deduce some implications that are interesting for the design of collaborative artificial agents. For example, their first observation was related to MOO dialogues or whiteboard items. They noticed four categories of content:

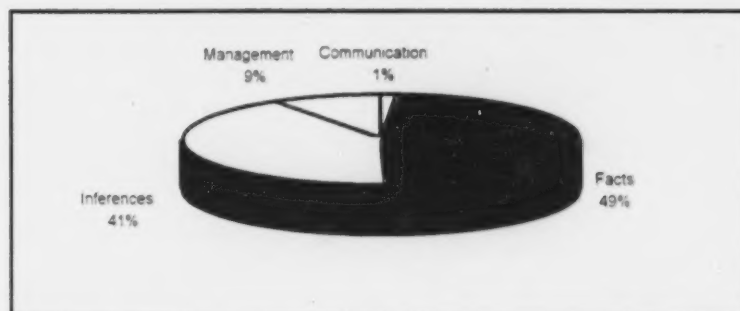
1. task (subdivided into facts and inferences),
2. management (who has collected what, who goes where, who is where, etc.),
3. technical problems and
4. meta-communication (turn taking rules, graphical codes, etc.).

The distributions of these content categories differs between MOO dialogues and whiteboard notes, as can be seen in Fig. 8.a) and 8.b). In dialogues, 33% of interactions concern task management, versus 9% on the whiteboard. Task management, technical problems and meta-communication utterances have a short term validity, while task level information is more persistent, especially facts: the former categories represent 45% of MOO dialogues (semi-persistent) versus only 10% of whiteboard items (persistent). Moreover, the average delay of acknowledgment is shorter in MOO dialogues (48 seconds) than in whiteboard acknowledgment (70 seconds).

This difference probably reflects the difference of persistence of these two media: there is no urgency to acknowledge information which will (probably) remain a long time on the whiteboard. Two forms of persistence of knowledge are mentioned: persistence of validity, i.e. how long some piece of information remains true and persistence of display, i.e. how long it can be viewed on a particular medium. The latter depends on the medium: MOO information is semi-persistent, it scrolls up every time a new command is performed; whiteboard information is persistent until explicitly erased.



a)



b.

Figure 10 Categories of content in MOO (a) and whiteboard (b) interactions

Implications for designing software agents are derived by the authors. First, they believe that an intelligent agent should select a mode of communication adequate to the persistence of knowledge being discussed. Negotiating non-persistent information through a persistent medium leads to the display of invalid information (when its validity has expired - a well-known WWW problem). Vice-versa, negotiating persistent information through a non-persistent medium increases the memory load, since the medium does not play its role of group memory.

Other observations are presented and they are related to: reasoning on sharedness, reasoning on mutual position, negotiation by action, dynamically allocate functions to tools etc.

Example 1: Associate agents in virtual environments are described in Appendix 6.

Example 2: A Prolog collaborative agent is briefly presented in Appendix 7.

Assignment 4. Connect to the MOO mystery solving game and solve it together with the artificial agent (HercuLOG)

4.6 WWW and education

The „World Wide Web“ (WWW) is a distributed hypermedia system that runs over the Internet. The World-Wide Web was conceived by Tim Berners-Lee at CERN in 1989 as an information integrator within which all available information on the Internet could be accessed in a simple and consistent way on every kind of machine architecture. A standard WWW browser (i.e. the client program for the WWW) can access at least the following communication protocols: HTTP (WWW's Hypertext Transfer Protocol), FTP, NNTP, WAIS and Gopher (see Figure 11 as presented by Schneider). Within some limits, a WWW browser can also launch local applications with data retrieved from a server. Finally, the „forms“ interface allows in principle access to any kind of external program running on a server.

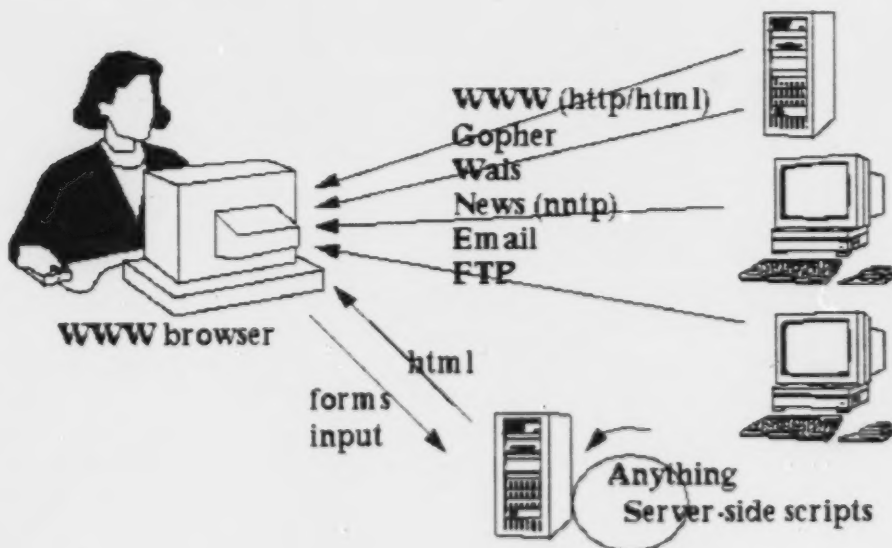


Figure 11 WWW browser

The WWW can be characterized with several functionalities (Schneider and Block, 1995):

1. A Knowledge Integrator
2. A distributed Hypertext
3. An Interface to any kind of remote program
4. An Interface to certain local programs.

The same authors distinguish several levels of WWW Use in Education. In order of difficulty they are:

1. The Web as information tool: Curricula and courses information, etc.)
2. Distribution of learning material: E.g. books (in various formats like postscript, word-processor, etc.), programs, applications needing a specific interpreter/language for the client
3. Collaboration tools
4. Interactive educational applications:

Recently a few interesting new developments can be observed in the WWW world, like: HotJava (a WWW client that can execute programs), WWW-MOO interfaces (a Moo-Web interface is described in Appendix 9), VRML (virtual reality mark-up language) interfaces, Netscape's „server-pushes“, Educational packages for CGI-scripts. These extensions and additions will add additional power to the WWW and increase its popularity.

Schneider and Block (1995) consider that five main aspects need to be considered before planning involvement with WWW technology:

1. Technological aspects (e.g. handling filters for generating HTML, Server-side programs, Data-Bases, etc.)
2. Organizational aspects: Who is going to do the extra-work? How can we train all involved persons?
3. Pedagogical/psychological aspects: How can we produce good course-ware on the Net? How can we build full teaching and learning environments?
4. Media aspects: How do we „mediate“ existing educational content? (This is an important issue for Distance Teaching institutions)
5. Cost: Internet access is expensive (rural areas!), producing WWW material can be quite time-consuming (including instructor's training), the students themselves do need some basic „Internet training“ too.

4.7 Intelligent interfaces for learning support systems

Introduction

As computer systems become more powerful and complex, our interactions with them have become more information laden and, consequently, more burdensome. It is now generally recognized within the human computer interfaces (HCI) and intelligent user interfaces (IUI) communities that as systems become more complex, this need for higher-bandwidth interfaces should be addressed by learning about and adapting to the user. The pieces to this puzzle are coming together from a variety of disciplines, including machine learning, user modeling, intelligent tutoring, information retrieval, and data mining. Furthermore, related work is discussed in the field of autonomous agents.

Solutions to the above mentioned problem are given by the Artificial Intelligence community as well as the Interface Design community: interfaces that incorporate some artificial intelligence as well as interfaces that constitute „an intelligent design“.

When designing a learning interface, several issues will have to be considered:

Problem Domain:

- what is the task?
- what is unique about the task and why is it important?
- what will a solution in this domain tell us about general solutions?

Approach or Method:

- how was data (from which to learn) collected?
- what learning algorithm was used?
- was learning on-line or off-line?
- how was the learned model utilized?

Evaluation:

- how do you measure success on the overall task?
- how do you measure improvement for a given user?
- what were the causes for success or failure?

The WWW and Internet interfaces have to provide greater support for students and teachers. Such support might involve user modelling, monitoring of mistakes and progress of the students, as well as the possibility of richer graphical environments (WWW6 workshop, 1997).

Many of the WWW technologies that now exist would be used to build more sophisticated interfaces, e.g. Java, VRML or even some of the more exotic plug-in formats. However as with any fast moving technology it will take time before the applications reflect the power of what is technically possible.

The standard WWW information vehicle is a hypertext document, a text file encoded with the so-called „Hypertext Mark-Up Language“ (HTML). Forms are an important feature of HTML. They allow a server to query the user with a few standard graphic user interface (GUI) widgets like push-buttons, radio-buttons, text editing window and scrolling lists. This information can then be processed by the server. It makes up a powerful query-interface to various kinds of data bases, but it can be used as interface to any program running on a server (WWW6 workshop, 1997). Because HTML pages can be generated dynamically by a server, such pages can be tailored according to the needs of the user, which is an interesting feature for educational systems. Another feature is interactive maps which report to the server the position of mouse clicks. Such maps can be used to build user navigation aids.

Examples

1. Adaptive Human Computer Interface for Improved Learning in the Electronic Classroom - Investigation of the Effects of Adaptive Information Access Modes on Computer-Based Training (CBT) (TECH REACH Inc.)

The concept involves recognizing and adapting to student's *information access mode preference* to meet the information gathering needs of students (e.g., emphasizing aural, visual, or kinesthetic

interaction and remediation –Figure 12). This can be used in a stand-alone environment and electronic or traditional classroom environment, involving individual as well as groups of students. The objective of this Phase II SBIR is to quantitatively test the hypothesis that computer assisted training, which adapts to the information access needs of individual students, will significantly improve training effectiveness while reducing training time and costs. A prototype CBT system is being designed, implemented, tested and evaluated in three formats (aural, visual, and kinesthetic). A commercial test instrument is also under investigation, which will accurately assess a student's most effective learning mode, based on information from the test instrument and personal preference.

Adaptive Learning Concept . . .

Figure 12 Adaptive human computer interface

This project developed for the US Navy will result in an automated learning assessment tool having widespread application to improve training for Navy personnel in stand-alone individual learning environments (Learning Resource Centers, web-based, etc.), group-paced electronic or traditional classrooms, and for the commercial training market.

2. Belvedere system

The Belvedere system is an architecture that places shared resources and „heavyweight“ functionality on servers, and uses Java and Netscape to deliver student interfaces on a wide variety of client platforms at any location with Internet access (Suthers and Jones, 1997). The representations used build on existing standards, embedding semantic annotations that support advanced functionality in materials that are also accessible to more conventional software. The implemented system includes groupware and associated tools that support students engaged in critical inquiry processes, such as investigating a scientific problem.

The Belvedere „inquiry diagram“ interface (Figure 13) can be thought of as networked groupware for constructing representations of evidential relations between statements. It uses shapes for different types of statements and links for different kinds of relationships between these statements. Multiple clients can view the same inquiry diagram, with „what you see is what I see“ (WYSIWIS) updating. An auxiliary „chat“ window (upper left of Figure 13) supports unstructured natural language communication. Additionally, a software-based „coach“ (lower right of Fig.) provides assistance to students as they engage in their various inquiry

activities (Paolucci, Suthers and Weiner, 1995) and (Toth, Suthers and Weiner, 1997). To avoid interrupting students' thought processes, the coach is minimally intrusive, usually remaining quiet unless students ask for advice, and flashing its light bulb only when it has critical advice to offer. It coaches critical inquiry by asking questions students may not have thought of, based on criteria of inquiry and argumentation in science.

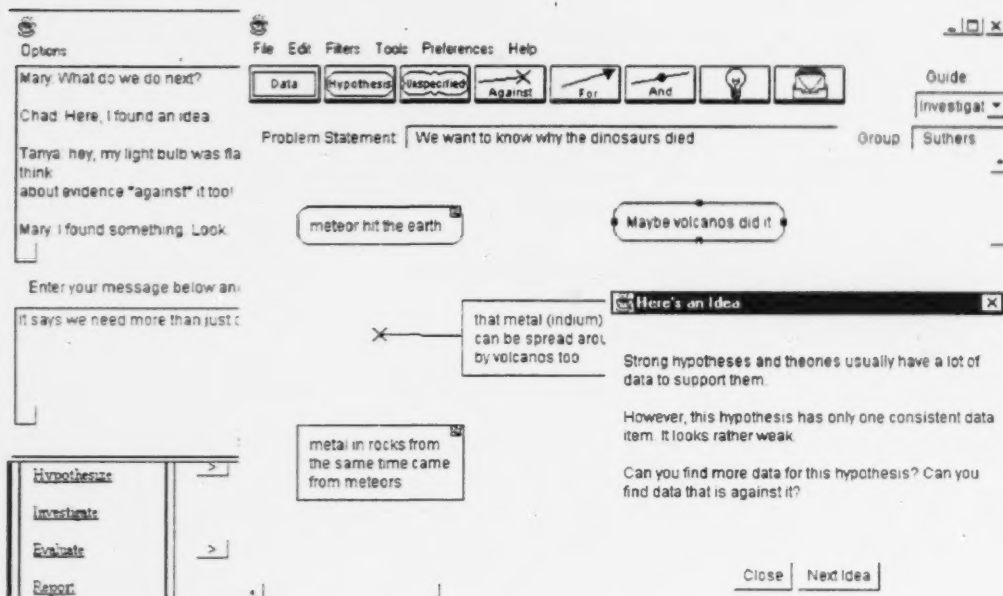


Figure 13 Belvedere system – a graphical interface for critical inquiry

3. Learning interface for collaborative problem solving

Research on and discussion of intelligent software agents has mushroomed in the past few years. A key area of research in this field is designing artificial agents able to collaborate with a human agent for solving a given task. A semi-structured intelligent interface that was designed in order to be used by the human agent for interacting with the a virtual world (MOO) and collaborating with the artificial Prolog agent is presented in Appendix 8.

The first and most important feature for this interface is that it is able to generate simple natural language propositions, translate them into a Prolog format and send them to the artificial agent. Conversely, the interface is able to recognize Prolog facts sent by the artificial agent and converse them into a simple set of the natural language in order to be understood by the human agent.

The interface is implemented in the Tcl/Tk scripting language and the structure of the whole experimental setup is shown below in

Figure 14.

BEST COPY AVAILABLE

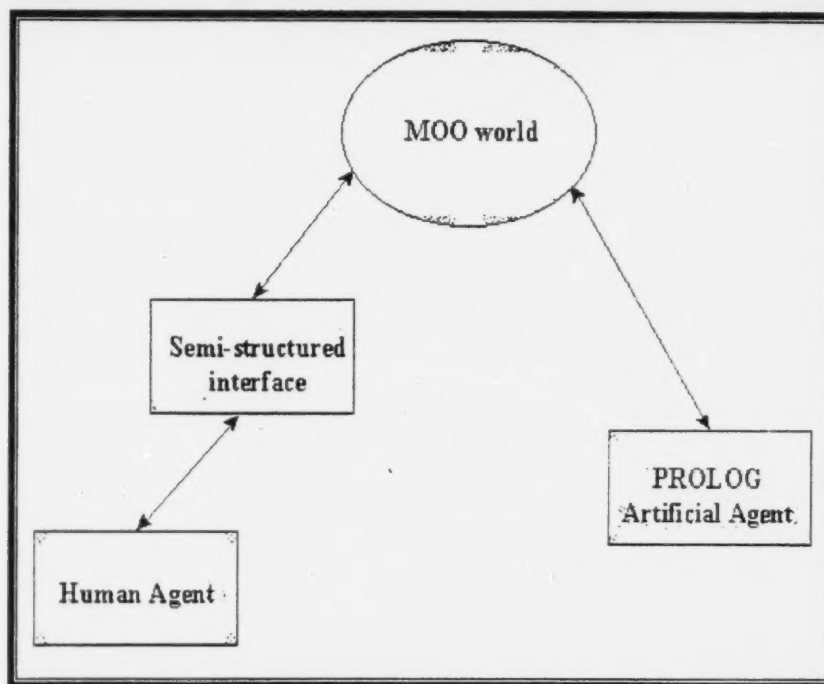


Figure 14 Collaborative problem solving experimental setup

The interface can be run from a web page using the TCL/Tk plugin for Netscape. More details can be found in Appendix 8.

Assignment 4. Connect to the interface and solve the MOO mystery game by using it for co-operating with the artificial agent

A step towards a truly learning intelligent interface for collaborative problem solving would be to design software agents capable of identifying a problem solving phase when observing the interface usage and interaction parameters. Such agents could then orient and supervise the usage of the interface in a way to facilitate the interaction. In a distributed cognition perspective, we don't model a student anymore, but a larger system including people and artifacts. To do so, we first have to identify characteristics of artifacts such as media persistency and correspondence between artifacts and knowledge types (See Dillenbourg & al., 1997). Quite appealing might be the use of observer - agents, non-intentional agents which consist of a recognising module which information is more or less displayed in a raw graphical manner to the collaborating subjects. Observers are agents that collect information regarding users interactions, aggregate observations into high level indicators and display these indicators to the human coach or to the users themselves. One cannot a priori set up collaborative settings which guarantee effective collaboration, hence a coach must monitor the interactions (Dillenbourg et al, 1995). Observers would support this monitoring process both for human and artificial coaches. These indicators could also be shown to the subjects. We intend to study how a pair could use them to regulate its interactions. We are then close to another approach in which shared external representations are designed to replace intelligent system advice. For instance, Ploetzner et al. (1996) study problem-solving in physics and use a computerised tool which allows subjects to collaboratively build concept maps. These external representations have been designed in order to help students to link qualitative and quantitative knowledge. A major question is whether the socio-cognitive behaviour of the peer changes when it gets information about its interaction. First experiments with observer agents were reported in (Buiu, 1997).

5. Future perspectives

„Computers in the future may weigh no more than 1.5 tons.“

Popular Mechanics, forecasting the relentless march of science, 1949

„640K ought to be enough for anybody.“

Bill Gates, 1981

As you can see from the above quotations it is quite risky to make predictions about the future of AIED. However, we cannot resist the temptation of doing so.

Education today has to play very actively its role in the society. The educational systems have and will have to respond to the pressing economic and educational problems of the human communities. The youth has to be inspired, guided and trained towards a global vision.

Technology mediated learning will become, very rapidly in our opinion, the vehicle for fulfilling the needs of Lifelong Learning (LLL). Various new models of education are evolving around the globe (especially in the U.S.) in response to the more and more pressing challenges of LLL.

As (Beller and Or, 1998) mention, traditional learning can no longer satisfy all learning needs, for the following reasons:

- (1) High quality learning depends, to a large extent, on finding a sufficient number of suitable lecturers;
- (2) Studies in public research universities are expensive; thus, accessibility is usually limited and subject to budget cuts and restrictions;
- (3) Traditional learning is limited to a particular place (the classroom on campus, which is also expensive to set up), a specific time, and a uniform pace.

The cost of education is dramatically increasing and some individuals even believe that „Thirty years from now, the big university campuses will be relics. Universities won't survive... Do you realize that the cost of higher education has risen as fast as the cost of health care?“ (Forbes Magazine, 1997, pp. 126-127).

So, models of complementing the traditional residential higher education are being proposed all over the world. Today's educational system is a hybrid one as traditional face-to-face education („sage on the stage“) coexists with high technology mediated learning support systems („guide on the side“). Lecturing and telling is very slowly replaced by facilitating and guiding.

Artificial intelligence is still a dream for most of the computer scientists. However, many AI techniques can be used in implementing high performance learning support systems. The role of „true“ AI in education is increasing very slowly. Machine learning, belief revision, frame theory, ontologies, diagnosis, planning, non-monotonic reasoning are studied by AIED researchers in order to support the design of learning environments.

Topics of AI that we believe that will strongly influence the future of AIED field are:

1. Machine learning: it is applicable to many aspects of AIED, such as adaptive tutoring techniques, student modelling and providing education partners for a student. More theoretical research is needed in developing different and more efficient learning strategies.

2. Ontology engineering: „ontology“ is expected to serve as the new, strong foundation of knowledge engineering. The idea is that a deep understanding of „content“ will give us new insight into design of knowledge representation. The notion of „ontology“ can be key to this issue. The ultimate goal of research on ontology is to give the full picture of theory of knowledge. To make improvements in the study of this difficult issue, of course, it is important to accumulate huge amount of „contents“, and develop sophisticated ontology representation language as fundamental „form“ of knowledge (Mizoguchi and Bourdeau, 1999). The same thing applies to the field of intelligent educational systems (IES). Building an IES requires a lot of work. At the present situation, however, it is always built from scratch.
3. Natural language processing: research in this area is expected to provide more performant human-computer interfaces and new, extended capabilities to artificial agents;
4. Software agents: the human tutors will disappear completely from the stage in the next 10 years. Instead of them, the human learner will be assisted by collaborative software agents („cobots“), by synthetic characters (animated and conversational artificial characters), by chatterbots etc.
5. Authoring tools: these are software tools that could decrease the cost of building intelligent learning environments, allowing more people to participate in building them. We believe that developing authoring systems to build intelligent tutors will be driving force of revitalizing the field of Intelligent Tutoring Systems.
6. Educational robotics: it involves mainly the use of computers to acquire, analyze, control, and model different physical worlds with real device control. We think that 1 (physical experiments, micro-robots, flexible workshops) will dominate the education in 25 years from now.
7. Educational computer games and environments: we believe that the education of the future will be very much devoted to develop skills in the early phases of the life. Educational games add „engagement“ and „pleasure“ to learning. These are key factors in improving the motivation and efficiency of the learning environments. We believe that this field will „explode“ in 8-10 years from now.

As we remember from the chapter on defining AI-ED, one of the definitions said that [Mizoguchi, Ikeda and Sinitsa (1997)] „Artificial intelligence in education (AI-ED) research consists of the following three major research areas in addition to Computer Science: Artificial Intelligence techniques and theories;

4. Artificial intelligence;
5. Educational technology;
6. Cognitive science.“

Improvement in the educational technologies and cognitive science will very much influence the future of AI-ED.

1. Cognitive science: we believe that the incredible simple processes that make us intelligent will be better understood in the next years and that new, more powerful, computational models of cognition will be developed. We think that the operation of the brain will be better understood in 10 years from now. However, as this „decade of the brain“ (1990's) has suggested, consciousness will remain a mystery for a long time, unless science will rediscover the ancient theories of consciousness and techniques for self knowledge. We strongly believe that computers will not become, by no means, conscious as they do not have a Self. An artificial Self will be impossible to be designed in our opinion. Let us remember what Einstein said a long time ago:

“Science without religion (spirituality) is lame; religion (Spirituality) without science is blind“. However, we think that there is a big chance for science to become spiritual (we can think on modern quantum physics which rediscovers ancient theories of spirituality). In our oppinion, rediscovery of ancient schools of spirituality, such as the prominent Kashmir Shaivism school of India, will give science a chance to rediscover its true foundations and goals.

2 Educational technology: we are facing nowadays impressive technological developments. Here are some areas that we think that will shape the future of AIED:

- Virtual reality;
- Digital communications;
- Internet and WWW;
- Multimedia.

Appendixes

APPENDIX 1

GENERAL, INTRODUCTORY INFORMATION ABOUT THE INTERNET¹

"On the Internet, nobody knows you are a dog."

(From cartoon showing a dog using a computer [Steiner, 1993].)

Introduction

The Internet is the biggest computer network in the world. It consists of a large collection of computer networks of differing kinds which link the most varied sorts of machines with each other - from PCs to mainframes.

The Internet is an extraordinary network because it belongs to no-one and there is no central management. The individual networks which comprise the Internet are maintained and developed further on a local level (with, for example, the support of the government). There are, however, a number of organizations that monitor certain aspects or sections of the Internet but there is no central organization behind them.

Thus, there is an organization which looks at the direction in which the Internet should be heading: the *Internet Society* (ISOC). This organization consists purely of volunteers whose single aim is to promote the free exchange of global information by means of Internet technology.

The technical aspects of the Internet are regulated by the *Internet Architecture Board* (IAB). They design and approve new network protocols and applications which can be used on the Internet on a large scale.

Finally, the body which is responsible for the registration of all computers and networks that are linked to the Internet, as well as offering special consulting services to the participating networks, is called InterNIC.

The Internet has been around for more than 25 years. However, its incredible rise in popularity is a very recent phenomenon (of the last two to three years). The most important driving force behind this rising popularity is the WWW, which - when combined with a user-friendly and easy-to-use browser such as *Netscape* or *Mosaic* - is a very attractive medium to use.

The money being invested in the Internet by both the various governments and also businesses, could comfortably be called substantial (particularly in The United States). This is an indication that governments and companies are taking the Internet seriously and that it is going to play an important rôle in future (international) developments in all kinds of fields.

Internet Services offered

The Internet provides access to an unprecedented amount of information about the most various of subjects, as well as to a great quantity of software for the most various of applications. Moreover, there are several services on the Internet which can considerably facilitate finding this information and/or files. Besides this, there are all sorts of worldwide forms of communication possible, such as electronic post and keeping up with newsgroups. At the moment, the Internet's information and services are still mostly free to obtain and use but the chance is high that, in the near future, payment will have to be made for access and use. When this will actually happen depends on such things as how long it will take before payments can be made on the Internet in a safe way. There are facilities existing at the present time but these are not yet reliable and safe enough to allow intensive use.

¹ This information has been largely obtained from the NBBI WWW-service: <http://www.nbbi.nl>

In the following overview you will find a short account of the Internet's most important features. We shall begin with the possibilities for (finding) information and files:

- **FTP:**

FTP is an abbreviation of *File Transfer Protocol*. This protocol is a sort of language which enables machines to communicate with each other and makes it possible to connect to an external computer and then have files sent from this computer to your own machine (or vice versa). FTP makes it possible to exchange all sorts of files with every sort of machine - as long as the other machine also uses this protocol;

- **Telnet:**

Telnet is a communications protocol which can make a connection to a computer elsewhere, after which it is possible to work on this external computer;

- **Gopher:**

Gopher is a system for searching for information via the Internet. Gopher works with a simple menu screen for finding information and thus shields the user from the underlying search mechanisms. The information offered may be anywhere in the world but, in principle, the user will not notice this and therefore need not concern himself about where in the world the particular information he is looking for is located.

As far as presentation is concerned, Gopher is simpler and more sober than a service such as the World Wide Web but, on the other hand, Gopher enables a relatively quicker search in most cases;

- **World Wide Web:**

The World Wide Web (WWW for short) is a worldwide information system which can be approached via the Internet and which is based on Hypertext. A hypertext document is a text which includes so-called links which connect to other texts or text fragments, video or audio (extracts) or graphic objects such as pictures. Links are recognizable because they are displayed in a different way to 'normal' text - for example, underlined or in bold type - but a link can also be hidden behind a picture. WWW pages can be called up/found by using a so-called *Universal Resource Locator* (or URL for short).

As far as the communication possibilities are concerned, the following facilities are available:

- **Electronic mail:**

Electronic mail (or e-mail for short) is a simple way of exchanging electronic messages between two people (or more). The only thing you need to know about the recipient of your message is his (worldwide, unique) e-mail address. Up till 1995, E-mail was far and away the most frequently used Internet service, but has been surpassed by the World Wide Web.

Sending a message goes in much the same way as sending a 'normal' message by post, only much quicker. Another advantage of e-mail is that it is not bound to certain times: you can send a message whenever you want and the recipient can read it whenever it best suits them.

So-called 'mailing lists' constitute a special use of e-mail. These are forums in which discussions on a specific subject are held via e-mail;

- **Usenet News:**

Usenet News is a worldwide conferencing system that comprises thousands of discussion lists about specific subjects called newsgroups. There is a news group for just about every conceivable subject. This might be a serious subject (such as science) but it could also be a much more light-hearted one (such as food and drink). The newsgroups are arranged in a hierarchy, based on the newsgroup's subject (computers, alternative, business etc.);

- **Internet Relay Chat:**

Internet Relay Chat (or IRC for short) offers the facility of 'chatting' worldwide and with more than one user at a time. The 'chatting' takes place by typing in messages which the other participants see on their screens.

APPENDIX 2**A LIST OF WORLD WIDE WEB SEARCH ENGINES**

There are many Search Engines on-line on the Internet. These search engines allow a user to search for information in many different ways, and are highly recommended web search tools for the time being. The following list² will give an idea of the kind of the search engines that are currently available. Between brackets the URL of the service (which is needed to find and use it) is given.

- **Achoo!** (<http://www.achoo.com/>):
Achoo! is a new Internet Health Care Directory, modeled after Yahoo (see later on in this list), it is one of the most comprehensive search sites for medical information. Containing over 5,000 sites, users can browse by subject categories with this quick search vehicle;
- **Affinicast Agent** (<http://www.affinicast.com/>):
A new way to locate Web sites geared towards your personal preferences. After administering a short questionnaire about your preferences for Internet content and activities, Affinicast provides a set of specific suggestions;
- **AliWeb** (<http://web.nexor.co.uk/aliweb/doc/aliweb.html>):
The Archie-Like Indexing for the Web is part of the Web at Nexor, in the United Kingdom. Their database is a collection of document summaries written by their publishers and regularly collected by ALIWEB;
- **Alta Vista** (<http://altavista.digital.com/>):
This is the first search engine created by Digital Equipment Corporation (DEC). Alta Vista is a quick, responsive, and easy to use search engine indexing over 8 billion words found in over 16 million Web pages and over 13,000 news groups updated in real-time;
- **Bess** (<http://www.bess.net/>):
Bess, the Internet Retriever for kids, families and schools is a new breed of Internet service provider specifically designed to protect children and others from the sexually explicit and adult-oriented material proliferating on the Internet. At the same time, Bess provides Internet users with a simple point-and-click environment to facilitate exploration of the thousands of educationally valuable and entertaining sites of the Internet;
- **B.E.S.T** (<http://eyecatchers.com/eyecat/BEST/>):
Best Education Sites Today is a search engine dedicated to education. With over 10,000 URLs in its database, it is the most comprehensive source for education links on the Internet. Users can Search by keyword, or by the Topic List, or browse the Awards for extensive reviews of the hottest education sites of the month;
- **Clearinghouse for Subject-Oriented Internet Resource Guides**
(<http://www.lib.umich.edu/chhome.html>):
Here you'll find Web links arranged mainly in educational categories, such as the humanities, social sciences, and science;
- **Computer ESP Internet Search** (<http://www.uvision.com/search.html>):
This site contains one of the most comprehensive, organized, up-to-date collection of search forms to Internet store catalogs, business directories, magazine indices, newsgroup indices, and Web indices related to the computer industry. Easily search dozens of stores for price and terms;
- **DejaNews Research Service** (<http://www.dejanews.com/>):
DejaNews is a tool for searching Usenet articles. Allows searches through mountains of Usenet archives in seconds to find the information you need. Fill-out forms and „how-to“ guides help you target your search to get what you want;

² The information in this list has been largely derived from C. Steele's Web page about WWW search engines (<http://www.interlog.com/~csteele/newbie3.html>). See this page for a very comprehensive and up-to-date list of search engines.

- **Electronic Library** (<http://www.elibrary.com/>):
Launch comprehensive searches across this deep database of more than 1000 full text newspapers, magazines, and academic journals; plus images; reference books; literature; and art. Just type a query or keyword in plain English and The Electric Library will quickly and simultaneously search 150 newspapers and newswires, nearly 800 magazines and journals, 3,000 reference works, and many important works of literature and art. And every article, story and reference work is full-text. This is a pay per use service, but at this moment there is (still) the possibility for a free trial;
- **EXPOguide** (<http://www.expoguide.com/>):
EXPOguide is a database of over 5,000 trade shows and conferences worldwide. Users can locate shows utilizing our concept search engine, or via location, date and alphabetical indexes. EXPOguide also contains listings of vendors providing services to the trade show industry;
- **Find Newsgroups** (<http://www.cen.uiuc.edu/cgi-bin/find-news/>):
This is a simple tool for discovering Usenet newsgroups of interest. Just enter a single string and a menu of newsgroups whose names or brief descriptions (not articles) match the search string will be returned;
- **Findex** (<http://www.findex.com/search.htm>):
Findex is the definitive global directory of financial institutions and services. Highlights include a searchable index of worldwide banks, security firms, stock exchanges, venture capitalists and all financial media on the WWW;
- **FTP Search 95 v3.0** (<http://ftpsearch.unit.no/ftpsearch/>):
FTP Search is an excellent search engine for locating what files reside on which server. Users type in keywords or the name of the file they wish to find, there are even several configuration options (such as the operating system that you use) which can be toggled before an search is initiated. The result is a quick list of FTP servers, with the path of the directory, and location of the file, designed as a quick link that can be access at the press of a button;
- **HYTelnet v6.8** (<http://galaxy.einet.net/hytelnet/START.TXT.html>):
HYTelnet is designed to assist users in reaching all of the Internet accessible libraries, Free-nets, BBSs, & other information sites by Telnet, specifically those users who access Telnet via a modem or the ethernet from an IBM compatible personal computer;
- **Image Finder** (http://wuecon.wustl.edu/other_www/wuarchimage.html):
The Image Finder, a thematic index for a vast image archive at the University of Washington, makes it possible to search for certain images on the Internet. Users simply type in a query or browse through the available list of catagories;
- **INFOSEARCH Broadcasting Links(c)** (<http://www.xmission.com/~insearch/links.html>):
INFOSEARCH Broadcasting Links(c) is a comprehensive hypertext directory of broadcasting related sites on the World Wide Web;
- **Internet Business Directory** (<http://www.ibdi.com/>):
The IBD is a new search tool allows users to find local, regional, national, or international companies by name, city, state, zip, area code or type of business. With over 20 million listings, this service provides free searches and listings for businesses;
- **ListWebber II** (<http://www.lib.ncsu.edu/staff/morgan/about-listwebber2.html>):
Using a forms-capable World Wide Web browser, you can use ListWebber to search the archives of LISTSERV or ListProcessor lists and extract only the information you want. ListWebber provides the means for searching LISTSERV and ListProcessor lists while reducing the need to know their searching syntax;
- **MediaFinder** (<http://www.mediafinder.com/>):
Request free information from a searchable database of newsletters, magazines, journals and catalogs. More than 5000 listings in 265 subject categories;
- **NetGuide's Calendar of Events** (<http://techweb.cmp.com/net/calendar/cal.htm>):

This service provides an online calendar covering current electronic events. Areas covered include Online services, Internet-Related Conferences, WWW Events, and other Event Calendars;

- **Notable Citizens of Planet Earth: Biographical Dictionary**

(<http://www.tiac.net/users/parallax/>):

An online searchable dictionary reference which contains biographical information on over 18,000 people from ancient times to the present day. Information contained in the dictionary includes birth and death years, professions, positions held, literary and artistic works, awards, and other achievements;

- **OKRA: Net Citizens Directory Service** (<http://okra.ucr.edu/okra/>):

Contains over 800,000 e-mail addresses, and is constantly growing. Allows users to search its index for registered users, and allows users to submit their own database;

- **Purely Academic** (<http://apollo.maths.tcd.ie/PA/>):

Purely Academic is a database recently launched on the Web by a group of Students in Trinity College Dublin. It is a searchable database of Academic links, and links that are of interest to people involved in research;

- **SavvySearch** (<http://guaraldi.cs.colostate.edu:2000/>):

SavvySearch is an experimental search system designed to query multiple internet search engines simultaneously. With help of a Search Form users can indicate whether they'd like to search for all or any of the query terms, and indicate the number of results desired from each search engine. When a user submits a query, a Search Plan is created wherein the nineteen search engines are ranked and divided into groups;

- **SIFT / Stanford Information Filtering Tool** (<http://sift.stanford.edu/>):

SIFT allows users to conduct searches and submit key words which skims thousands of Usenet news messages to find stories of interest. This free service will also notify you via e-mail once the articles you've requested are available;

- **Telephone Directories on the Internet** (<http://www.buttel.com/tel/>):

A collection of pointers to national and regional telephone directories on the Internet. Includes links to various US Yellow Pages, as well as a few directories for other countries such as Australia and France;

- **The WWW Virtual Library**

(<http://info.cern.ch/hypertext/DataSources/bySubject/Overview.html>):

Another good place to start exploring if you have a particular topic in mind, the Virtual Library includes topical and geographical indexes to Web pages;

- **Whoopie!: Index of Audio and Video on the Internet** (<http://www.whoopie.com/>):

A comprehensive audio and video search engine on the Internet. Live daily program guide of streamed audio and video. Allows a user to search both directories at once, individually, or browse through a number of categories including news, sports, medical, miscellaneous clips and educational documentary;

- **Yahoo!** (<http://www.yahoo.com/yahoo/>):

Created by David Filo and Jerry Yang from Stanford University in March 1994. Organized and structured using menus, instead of user prompts. Very easy to use, and quick response time, this site is the prime and most favourable location for web links for many users;

- **Yellow Pages & Web Page Search** (<http://superpages.gte.net/>):

An online Yellow Page site which has a good search capability for 10 million yellow page listings and 50,000 Web sites.

APPENDIX 3**MOO BASIC COMMANDS****TecfaMOO**

Here is some basic information on how to do the most basic actions in a MOO. You can always type 'help' to get an index of help topics. To get specific help about communication, for example, you can type 'help communication'.

MOOs are organized around „rooms“, i.e. you are always in some location. So a first thing you need to learn is how to look at the room you're in: Just type 'look'. To get information on the objects around you, you can type 'look <object>'.

To communicate with people in the same room you're in, you can „talk“ (use 'say') to them by typing something like:

say Hello, there.

- or -

„Hello, there

In order to address yourself publicly to a person in the same room, use a „-“ (dash), as in the following example:

-Daniel what did you mean ?

You say to Daniel, „what did you mean ?“

To know who else is connected at the time, type '@who'. You get also information on where they are, for how long they have been connected, and when they were active last.

To communicate with people not in your room, you can 'page' them like in the following example:

page Jane Do you have some free time ?

- or -

'Jane Do you have some free time ?

If you want to go to another room, you can walk through an „exit“ by typing the exit's name. Exits are normally listed when you 'look' like in the following example:

>look

Daniel's Office

You see open boxes, a desk and other brand new office furniture

Obvious Exits: Atrium (to The Tecfa Atrium) and Tower (to Tower).

Finally, to log off, type '@quit'.

Also note that in TecfaMOO most „interesting“ places are linked by the „Underground Corridor“ (UC). If you are lost, you can always type 'home' which will bring you back to the Arrival area.

Registered users ONLY can use the '@join' command to join a person as in the following example:

>@join daniel

You join Daniel.

LINGUAMOO

MOO Introductions:

First--Describe and Gender Your Character:

type '@describe me as [whatever you want to look like]' EXAMPLE: @describe me as A wandering rhetorician interested in third sophistic rhetorics, contemporary critical theory, and racquetball.

type '@gender'

a list of possible gender choices will appear. Pick one.

type '@gender [whatever]' EXAMPLE: @gender royal

type 'look me'

you'll get a description of your techno-body according to what you described

type 'look'

you'll get a description of the room your in and the people who are there

type 'look [whoever]'

choose someone else who's in the room with you and look at them.

You'll get the description they have set for themselves.

Second--Move, Read, and Speak:

type 'ONE'

you'll teleport into workroom One. You'll see a note called Second there.

type 'Read Second' and follow the instructions it displays.

Third--Speaking Up:

When you have read Second, completed the exercise, and returned to the main seminar room, type: 'Sit big table' again.

When you try to speak while seated at the table, only those also seated at the table will be able to hear you. If you would like the whole room to hear you, you'll have to 'Speak Up.' To do that, you'll precede what you want to say with an 'SU' instead of a double quote mark. For example, if I wanted to say hello to the whole room now, from my seated position, I'd type: 'su hello everyone.' What you would see is this: Dr.Davis speaks up: „hello, everyone.“ Try that now.

Fourth--Emoting:

By default, characters on Lingua can display several emotions with simple verbs. For instance, to smile, simply type: smile. When I type 'smile,' everyone in the room sees: Dr.Davis smiles.

Other simple emoting verbs are: giggle, grin, hug [whomever], wink [whomever], laugh, laugh [whomever], etc. If I typed 'hug cynthia,' assuming Cynthia was in the room, everyone in the room would see: Dr.Davis hugs Cynthia. If I type 'laugh,' you would see: Dr.Davis laughs. But if I typed 'laugh cynthia,' you would see: Cynthia causes Dr.Davis to fall down laughing.

To display emotions on the MOO that are not pre-set, precede your emotion by a colon [:]. The colon is short for 'emote.' So if I wanted to display myself jumping up and down, I'd type ':jumps up and down!' and you'd see Dr.Davis jumps up and down!

Try some emoting now.

Fifth--Paging and MOO-Mailing

To page anyone on the MOO at any time, simply precede what you want to say to that person with the word 'page.' If Cynthia were in workroom One and I were in workroom Two, I could type: 'Page Cynthia Hi there!' Cynthia would see: Dr.Davis is looking for you in Workroom Two. She pages: „Hi There!“

Try paging one of your classmates.

To email someone on the MOO:

type: @send [whomever]

hit enter

at the prompt, type a subject line

hit enter

at the prompt, begin your message. Note that each line must begin with a double quote mark.

So begin with one, and then every time you hit return, start the new line with another quote mark.

when you're finished, hit return

type send

hit return again. The MOO will say: Mail actually sent to [whomever]

More MOO Help

For more help on MOO commands, type 'HELP' from anywhere on the MOO.

APPENDIX 4**END USER TAXONOMY OF SOFTWARE AGENTS****1. Desktop Agents**

1. Operating System agents: interface agent that provide user assistance with the desktop operating system
2. Application agents: interface agents that provide assistance to the user in a particular application
3. Application suite agents: interface agents that help users in dealing with a suite of applications

2. Internet Agents

1. Web search agents: internet agents that provide search services to users
2. Web server agents: internet agents that reside at a specific Web site to provide agent services
3. Information filtering agents: internet agents that filter out electronic information according to a user's specified preferences
4. Information retrieval agents: internet agents that deliver a personalized package of information to the desktop according to user preferences
5. Notification agents: internet agents that notify a user of events of a personal interest to a user
6. Service agents: internet agents that provide specialized services to users
7. Mobile agents: agents that travel from one place to another to execute user specified tasks

3. Intranet Agents

1. Collaborative customisation agents: intranet agents that automate workflow processes in business units
2. Process automation agents: intranet agents that automate business workflow processes
3. Database agents: intranet agents that provide agent services for users of enterprise databases.
4. Resource brokering agents: agents that perform resource allocation in client/server architectures

APPENDIX 5**MOO OBSERVER AGENTS (MOObservers)**

Observers are software agents who collect information regarding users interactions, aggregate observations into high level indicators and display these indicators to the human coach or to the users themselves.

Sample observer agents in MOO worlds are described below. You will need more rights in case of use of the MOO programming level. These rights can be request from the administrator of the MOO server, after you probe that you will respect the rules of the general behaviour. At the programming level you will use also the functions, which can be (re)defined. The (re)writing of the verb is a simple thing: you can introduce the code in the editor, after that you must compile the code (at this moment you can remove some errors) and you can use the new verb defined. At the programming level you need to know more about the structures of the objects. The objects have a general name and the MOO system gives a unique number to each object. Each object has the verbs (the code part) and/or properties (attributes - the data part). The „observer“ is the basic form of the agents which can be implemented in the MOO world. This agent can observe something and can transmit the observations. We have chosen two observable verbs: 'say' and 'page'.

The easiest communication commands between players are 'say' and 'page'. The 'say' command is used to say something in the room where the player is. All players in the same room can see this „something“. The syntax is the following: say <anything> where <anything> is something which is not in contradiction with the behaviors rules. You can use the 'page' command if you want to talk with somebody, which isn't in the same room. The syntax is:

page <name> <message>

where the <name> is some valid player name. Here you are some examples of the commands:

>say something to read

You say, 'something to read'

>page Cristian Example of page

Cristian has received your page.

Cristian has been idle for 1 hour, 2 minutes, and 31 seconds.

Another player will see:

Cristian says, „something to read“

You sense that Cristian is looking for you in My recreation room.
He pages, „Example of page“

It's interesting to have some statistic approach of the communication between players. This is the role of the MOObservers we have implemented.

The implementation of the MOO observers agents

The writing in a database is the main idea of the implementation of the MOObservers. The database can be statistically interpreted by a special verb created specially for this.

The verbs on the MOO aren't independent; they're attached from well-defined objects. For example the 'say' verb is attached on the room where it's used (so, each room has attached one 'say' verb), and the 'page' verb it's a player verb (so, each player has attached one 'page' verb). We have chosen the modification of the 'say' and 'page' verbs to fulfill the database. So we have chosen one room for the 'say' verb modifications and two predefined players for the 'page' verb modifications (with the players accept). The database was ready to be exploited: the number of characters, the time when the characters are recorded (the characters are in the <message> field), and the player which is on observation.

The accumulated data can be shown both in the text mode (one verb create specially for that) and in the graphic mode (one TCL interface was made for that).

Pointer - a very object

The generic object called 'pointer' is in charge with the information control (that information that are extracted from the observation of the use of the 'say' and 'page' verbs). The 'pointer' object it's like a database, which include in the 'users' property the names of the players that are monitoring. The 'users' attribute can be like following:

```
-----
>@d pointer.users
.users      Cristian (#4457) r c {#4457, #4447}
----- finished -----
```

where the @d operator (verb) means the show of the variable (the attribute of the object). The data are between the brackets; those numbers are the identifiers for the players. So we check two players with the identifiers #4457 and #4447, respectively. The object is owned by Cristian, which has the #4457 id.

The 'pointer' object has some verbs to control its properties modified by 'say' and 'page' verbs. They are the following:

```
-----
@d pointer:
Pointer (#4721) [ readable ]
Owned by Cristian (#4457).
Child of generic thing (#5).
Location Test (#4902).
1 #4721:instagent   Cristian (#4457) r d any out of/from inside/from this
2 #4721:statc      Cristian (#4457) rxd this none this
3 #4721:trimitc    Cristian (#4457) rxd this none this
4 #4721:checkagent Cristian (#4457) r d any out of/from inside/from this
5 #4721:stopagent  Cristian (#4457) r d any out of/from inside/from this
6 #4721:showp      Cristian (#4457) r d any out of/from inside/from this
7 #4721:com_page   Cristian (#4457) rxd this none this
----- finished -----
```


Verbs descriptions

The 'instagent' verb install the MOO agent. The syntax is the following:

instagent any from this

where „any“ is anything from {say, page, all}, „this“ is 'pointer' object, and „from“ is unchanged. This verb (instagent) call the 'statc' verb and exits. It will be signaled which agent was installed ('say' or 'page'), or if one of the agents was previous installed. The 'say' and 'page' installing is controlled by the '.stop_say' and '.stop_page' attributes of the 'pointer' object, respectively; the observer hasn't installed if the attribute has the 0 value and otherwise for the 1 value.

The ':statc' verb modifies the '.plotter1_say', '.plotter2_say', '.plotter1_page' and '.plotter2_page' attributes, which are attached to the 'pointer' object; those attributes are the players attached vectors in respect of the called verbs. The ':statc' verb it's an internal verb, so it can't be called from exterior (MOO prompt) (this is for the protection of the installing of the agent; the install has to be make only with the owning tools, not forced). The ':trimitc' verb is called after the completion of the vectors above; it take the information and display it. The ':statc' verb is reprogrammed to rerun by itself after n seconds (the number of the seconds is one variable of the 'pointer' object). The control of the auto run is associated with another variable of the 'pointer' object, which has the 1 value for permission auto running and otherwise for not permission.

The ':checkagent' verb (of the 'pointer' object) check and display information about the observers installations. In fact the '.stop_say' and '.stop_page' attributes are checked; it's usefully for know about the agents installing. The syntax is the following:

'checkagent any from pointer'

with the signification from the 'instagent' verb.

The ':stopagent' verb unload the agents, i.e. reset the '.stop_say' and/or '.stop_page' values. The syntax is the following:

'stopagent any from pointer'

with the signification from the 'instagent' verb. The '.statc_status' is resetting if both agents are unloaded: this is the way that the reruns of the "'statc' verb are stopped.

The ':showp' verb show the attributes that are involved in „observation“. The syntax is the following:

'showp any from pointer'

with the signification from the 'instagent' verb.

The communication with the TCL interface

The ':trimitc' verb use the XMCP/1.1 protocol to send the messages to the TCL interface, like the following string 'xmcp-say value44 time867756611' where the xmcp-say it's an identifier for the 'say' observer, value44 is the number of the character at that moment and time867756611 is the relative number of seconds from 1 Jan 1970. It's used some special driver, which is a 'pointer' object attribute. The internal used functions also something special: 'client_notify' and it is attached to the 'pointer.driver'. This verb is called in the same time with the reprogramming of the ':statc' verb, so the ':trimitc' verb it's controlled by the '.statc_status' attribute.

The TCL interface take the MOO information via XMCP/1.1 protocol, get the value and the time and show one graphic window, like functions graphics: the values versus time (the time is one calculus result: the difference between the time given and same reference time, which can't be bigger than 'pointer.tmax' seconds). The obtained information is including in same circular database, so the graphic will be moving inside the window.

An example of use of the MOObservers

One example of use of the observers is the following:

(the agents install)

>instagent all from p

SAY it's now installed.

PAGE it's now installed.

(the agents install check)

>checkagent all from p

'SAY' agent: Installed!

'PAGE' agent: Installed!

(the use of the 'say' verb: at the next time you'll observe the showing of one pixel in the 'say_obs' window, which is the number of the characters from „something...“ string)

>say something...

You say, „something...“

You say it!

(the use of the 'page' verb: at the next time you'll observe the showing of one pixel in the 'say_page' window, which is the number of the characters from „something...“ string)

>page me something...

You sense that Cristian [ceccone] is looking for you in Test.

He pages, „something...“

You page it!

(show all the parameters)

>showp all from p

Users: #4457 #4443

Times_say: 867753589 867753671 867753683 867753692 867753699 867753714

867753718 867753726 867753763 867753819 867754036 867755618 867755701

867755768 867755891 867756586 867756611 867757986 867757992 867759630

Players_say: #2571 #4457 #4457 #2571 #2571 #4457 #2571 #4457 #4457 #2571

#4457 #4457 #4457 #4457 #4457 #4457 #4457 #4457 #4457 #4457

Number of chars_say: 29 30 36 8 7 10 39 20 43 68 3 4 53 49 49 43 81 63 20 12

Plotter1_say: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 12 0 0

Plotter2_say: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Times_page: 866805376 866805748 866805758 866805758 866805813 866805836

866805858 867145603 867327064 867327086 867660016 867660037 867662414

867675256 867749693 867749714 867749893 867749936 867750215 867759647

Players_page: #4457 #4447 #4457 #4447 #4457 #4457 #4457 #4457 #4447

#4457 #4447 #4457 #4457 #4457 #4457 #4457 #4447 #4457 #4457

Number of chars_page: 19 100 19 23 24 25 25 8 100 37 100 34 34 21 14 46

119 100 43 15

Plotter1_page: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 15 0

Plotter2_page: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

(trying to reinstall)

>instagent all from p

SAY it's already installed!

PAGE it's already installed!
(agents unloading)
>stopagent all from p
Uninstall 'SAY' agent... OK!
Uninstall 'PAGE' agent... OK!

This monitoring is useful in observing the verbal flow in last 5 minutes, for example. If the discussion it's about something like predefined theme it can be show the discussion dynamic. This can be make off-line by logging data in one file and the file interpretation. This kind of statistical stuff can be extended to any MOO commands (the number of rooms visited in last 30 minutes, the number of questions, the distance between the player and one room, the number of typing faults, etc.). It can be done same map with the survived players and updating on-line. All this small programs assume the changes perception in the MOO world and transmit that to one interface.

Online demo

You must connect to the TECFAMOO world via tkmoo client
The character where the observers were tested is: babone
The password can be found at Catalin Buiu (cata@cib.pub.ro)
After connecting the commands are @xcmp_challenge for show the graphics and all commands mentioned above on installing, stopping and running the observers.

APPENDIX 6**CAETI CENTRE ASSOCIATES**

The CAETI Centre is a research effort, sponsored by Dr. Kirstie Bellman of the Defense Advanced Research Projects Agency (DARPA), investigating the design and use of virtual spaces for collaboration in technical communities. It is a 3D Virtual Reality Environment that has been enhanced to include MUD-like chat capabilities and intelligent agents that guide users and support evaluation. The virtual reality paradigm is extended to include analogous 2D and text representations of the Centre space. Visitors to the Centre can move effortlessly between the 3D, 2D and text versions depending on their needs and preferences.

Currently the Centre provides a cyber-space in which technologists can „exhibit“ the results of their research and development efforts. Exhibitors may talk with developers and users, integrate disparate products, demonstrate product use, display evaluation results, and provide video-taped testimonials. Similarly, visitors to the Centre can learn about and experience these products via a variety of multi-modal interactions. Educators can showcase successful examples of technology use, and research colleagues can communicate and collaborate with the tools provided to support these activities. Parents can explore the new technologies that their children will be using.

WHO are the Centre Associates?

The Associates are intelligent agents - a receptionist and several guides - who work in a virtual environment, called the Centre. The Centre currently houses the projects from the Computer-aided Educational and Training Initiative (CAETI) funded by DARPA. Each room in the Centre has information and resources for each of the CAETI projects.

The Associate system delegates tasks to the multiple agents who can reason, plan, and communicate with visitors to the Centre. The Receptionist manages visitor traffic. She greets visitors, gives orientations, and arranges tours. Guides can give pre-packaged tours organized around certain themes or customize a tour based on the visitor's interests. The Receptionist and the Guides can suggest activities, answer visitor questions and respond to requests.

The Associate system is based on several key technologies: pattern-matching for chat with visitors, a Blackboard architecture for control, a rule-based system for reasoning in the knowledge sources, persistent storage and support for communication. These technologies are described in more detail later.

Visitors can enter the Centre either through a 2-D or 3-D portal via the Internet. Visitors and the Guide Agents exist simultaneously in both the 2-D and 3-D environments (they see the same things, can chat with each other, etc.). The only difference is that 3-D version has graphic room representations and avatars for the visitors and the Guides. The 2-D world currently has no avatars and has text representations for the rooms. The agents are integrated in this distributed system with components developed by other Centre team members -- a multi-user object oriented (MOO) environment (Wyndhaven MOO from Intermetrics) and a commercial virtual reality modeling language (VRML) graphical server component (from Black Sun) for the 3-D world by UTA.

This new Centre is open to the public for beta testing at: <http://www.centrevue.com>

WHAT does the receptionist do?

Greets the visitor by name and offers an orientation or a guided tour.

Gives an orientation to the Centre.

Answers general questions about the Centre, such as

What can I do in the Centre?

What is the purpose of the Centre?

Tell me about CAETI.

Arranges a guided tour of the Centre.

Maintains a list (queue) of visitors waiting for a tour.

WHAT does the guide do?

Interviews the visitor to determine interests and time available for a tour.

Plans a customized tour that fits the visitor's interests.

Takes the visitor on a guided tour through project rooms in the Centre.

Answers questions about the Centre, its projects and some general topics, such as:

Who developed the Centre?

What is a MOO?

What is an Associate system?

Who developed this project?

What is different about its technical approach?

Suggests actions and activities, such as:

Recommends specific resources within project rooms to check out

Reminds visitors of time remaining on tour

Responds to visitor requests, such as

Take me to the next room. What should I do now?

HOW do the Centre Associates work?

> The Receptionist and Guides are intelligent agents based on Associate technology. This technology includes:

a knowledge base containing information about the Centre and the projects

control mechanisms for planning and managing events (BBK Blackboard)

rule-based reasoning for general knowledge and dialog management (M.4)

pattern-matching for understanding visitor queries and requests

communication among software components (KQML)

> If you visit the Centre, be sure to ask the receptionist and guides how they work. They can answer questions, such as:

Are you real?

How do you understand me?

What are intelligent agents?

What is KQML?

WHAT IS NEW or different in the technical approach?

The primary contribution of this work is its open architecture and systems engineering of components for multi-agent associate systems. The Receptionist and the Guides can be simultaneously active in the Centre, each accomplishing different tasks by reasoning and planning and communicating with visitors and with each other. This is different from most single purpose agents on the Internet, that tend to be designed to do just one task well, such as search for information.

PRACTICAL INSTRUCTIONS

Your new CAETI Centre character has been created, with name „siva“ and password „yxarimu“. (Passwords are case sensitive.)

„siva“ will start out in the CAETI Centre Lobby. You will be in novice mode to start with. The communications applet where you communicate with other players and type commands is the frame that spans the bottom of your Netscape window. It consists of an input buffer at the bottom and an output buffer where you will see the results of commands you type and communications from other players.

The novice capabilities are as follows:

By default, everything you type is broadcast to others in the same place. For example, if you type 'Hi everyone', you and anyone else in the same area or room will see the message that you said, „Hi everyone“.

There are three exceptions to this rule:

1. If you type „help“, you will get helpful information about the system. (If you actually want to say help, rephrase your sentence so that „help“ is not the first word.)
2. Text prefixed with the „\“ character is passed to the system to be interpreted as a command. Here are some useful commands that you might want to use:
 - \who will show you everyone logged on to the system
 - \join <player> will move you to the same place as <player>
 - \follow <player> will move you whenever <player> moves
 - \unfollow <player> will stop you from following <player>
 - \@password <old password> <new password> will change your password.
3. Typing the word „expert“ will switch you from novice mode to expert mode. In expert mode, what you type is interpreted as standard commands in the environment. Typing „help“ in expert mode will give you information about all the commands that are available. To return to novice mode type the command „novice“.

Visit the Centre at http://moo.intermoo.com:8080/caeti_current

APPENDIX 7**HERCULOG – A PROLOG COLLABORATIVE AGENT**

Computational Logic systems (Logic Programming, Constraint Logic Programming, and their extensions, collectively referred to as „(C)LP“ systems) can offer an attractive environment for developing Internet applications (Tarau et al., 1996). On the one hand they share many of the attractive characteristics of popular network programming tools, including dynamic memory management, well-behaved structure and pointer manipulation, robustness, and compilation to architecture-independent byte-code. However, in addition, (C)LP systems offer some unique features such as very powerful symbolic processing capabilities, constraint solving, dynamic databases, search facilities, grammars, sophisticated meta-programming, and well understood semantics.

The Prolog language is one of the most successful AI languages since it supplies built-in mechanisms such as backward chaining search, backtracking, unification and dynamic database handling. Several AI tools have been built using Prolog such as Flex (Quintus, 1991). These tools provide a set of high level knowledge representation mechanism and a set of associated deduction methods. One more reason for our option is that the Prolog is goal oriented and can follow different traces of inference for solving one logical problem.

HercuLOG – a PROLOG collaborative agent

Given the mystery game previously described (Assignment 3), we have designed two identical artificial agents (HercuLOG and SherLOG - Hercule and Sherlock written in BinProLOG) able to collaborate for solving the mentioned task. They are implemented as a rule-based system written in BinProlog and interfaced with the MOO environment via (fig. 1).

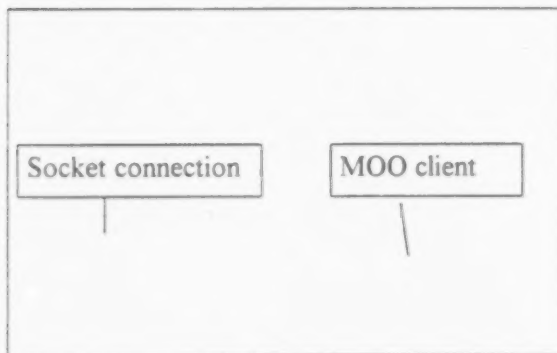


Figure 15 The structure of the collaborative problem solving system

We did experiments with 2 human agents solving together the mystery, with 2 artificial agents that co-operate for solving the task together and with a human plus an artificial agent. First, HercuLOG solved the mystery alone using a rule base consisting of reasoning rules.

The first partner of HercuLOG was the other 'artificial' agent named SherLOG. Both agents begin the investigation (after the socket connection to TecfaMOO) with the same known facts. The rules in the both case are more numerous than the previous version (HercuLOG only), because there are rules for the coordination part (decisions, suggestions, questions-answers). The duration for solving the mystery in case of HercuLOG working alone it's about 15 minutes. When both Prolog agents are working the time duration for solving the mystery, after the communication is established, it is around 12 minutes. The agents can be observed live in a *monitor* window, which is another MOO

connection, specially made for that. The log which is resulting can be interpreted off-line for seeing the *page* commands (communication part between agents) or *walk*, *look*, *read* and *ask* commands (investigation part).

The second partner of Herculog was Sherlock, as a human agent. The number of rules is more increased, because of variety of the decisions that the human agent can take, the questions that the human agent can ask, the answers that the human agent can give.

The Prolog agent works in the main cycle that it is shown in Table 3.

Table 3 The Main Cycle

Step	Action
0	Connect to the MOO
	Establish the communication with the other agent
	Initialize the game (only one from these two agents)
	Read from the MOO (via socket)
1	Think (applying the rules)
2	Send information to the MOO (<i>advice</i> from the other agent, the movement command – <i>walk to</i> , or issue the commands from MOO – <i>look</i> , <i>read</i> , <i>ask</i>)
3	Read from the MOO (via socket)
Repeat	steps 1, 2, and 3 until the mystery will be solved (i.e. the agent find the criminal)
4	Report who is the criminal
	Disconnect from the MOO

The Prolog agent has the following things to do: to keep the MOO connection alive, parsing the messages from the MOO, creating the personal facts base and making the logical connection between facts.

The parsing of MOO messages was a little bit complicated. It was proposed a structure of the messages like following:

#<Prolog_predicate>#<Prolog_predicate>...#<Prolog_predicate>

where the '#' character precedes only one Prolog predicate (one fact is equal with one Prolog predicate). The agent extracts the Prolog facts, verifies the validity of them and asserts them in the facts database. In this case the messages are arrived from the MOO mystery game (suspects' answers or information about different objects).

The Prolog agent can reason based on facts database and give some intermediary solutions or hints to its partner (either another Prolog agent, or a human agent). The intermediary solutions can be some decisions to continue the investigation, some suggestions to make to the partner or some

questions to ask to the partner (the agent use the MOO 'page' command to communicate with the partner). The messages from the partner were structured also based on Prolog predicates, like in the following:

`$<Prolog_predicate>$<Prolog_predicate>...$<Prolog_predicate>`

where the '\$' character precedes only one Prolog predicate. In this way the Prolog agent knows about the source of information: MOO environment or the partner. In this case HercuLOG assert the facts only after a thorough verification and validation.

The Prolog agent can be interrogated about its number of present facts. The number of facts is around 500 at the end of the task in good condition. In the both agents case the number of facts can grow up to 600. Some graphical statistics is presented in Figure 16:

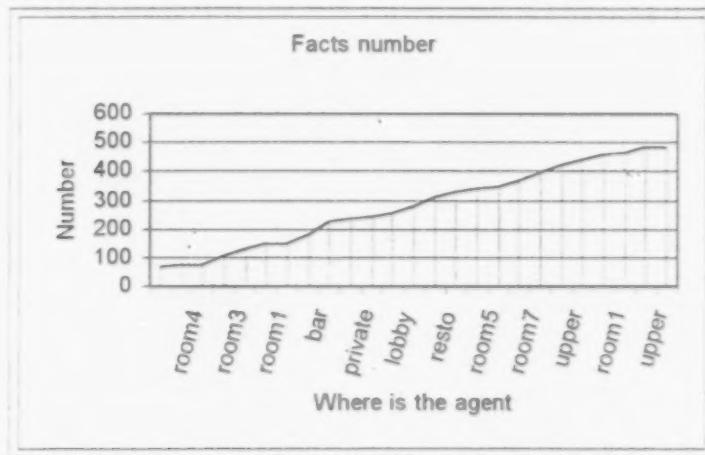


Figure 16 The evolution of facts number in case of HercuLOG solving the problem alone

As to the communication between the human agent and the Prolog agent, we can say that the human agent is using a semi-structured interface written in Tcl/Tk (see Appendix 9) and his commands are converted into Prolog predicates which are sent to the artificial agent. Conversely, the Prolog agent is sending Prolog predicates to the human agent, predicates that are converted into natural language propositions.

APPENDIX 8**INTELLIGENT INTERFACE FOR COLLABORATIVE PROBLEM SOLVING**

We will describe briefly a semi-structured interface for collaborative problem solving. Our problem (task) is the same MOO mystery game that is presented in Assignment 3. We have described an artificial collaborative agent able to solve a mystery game (see Appendix 7). The artificial agent is implemented as a rule-based system written in Prolog and interfaced with the MOO environment via sockets.

Waiting for Sherlock... Get Help Now Begin

Identification Name: MOO Connect Login Exit Help!

MOO Command: Send

Investigation commands: something something something something

Any command: Nov 25 17:43:19 E. Europe Daylight Time

Say: What's new? Page Of

Said: What's new?

HERCULE: Status disconnected

OK More info I know that Why? No Help (less than half of them)

Hercule's answers & things about...

Figure 17 A Tcl/Tk interface for collaborative problem solving

This appendix describes a semi-structured intelligent interface that has to be used by the human agent in order to interact with the virtual world (MOO) and to collaborate with the artificial Prolog agent (see Figure 17). The interface was implemented in Tcl/Tk a powerful scripting language.

So, the first objective for this interface is to generate simple natural language propositions, translate them into a Prolog format and send them to the other agent. Conversely, the interface will have to

recognize Prolog facts sent by the artificial agent and converse them into a simple set of the natural language in order to be understood by the human agent.

More details on the practical implementation of the interface can be found at:
http://surya.cib.pub.ro/cata/dcs/mod_5_04.html.

APPENDIX 9**MOO-WWW INTERFACE AT TECFAMOO**

TecfaMOO Arrival Station



Official Opening: around easter
Builders and Programmers Permits
will be issued

ECFA MOO Arrival Area

A Virtual Space for Educational Technology, Education, Resarch & Life
at TECFA, School of Psychology & Education, University of Geneva

'go w' (for Visitor Information) 'go s' (Builder's Information)

More information: <http://tecfa.unige.ch/tecfamoo.html>

You see Strange black dog here.

Guest is here.

Obvious Exits: Up (to Les Puces de Plainpalais), UC (to Underground Corridor), W (to Visitor Information), Up2 (to Transportation Center of Rond Point de Plainpalais), and S (to Underground Construction Office).

>@go atrium

The Tecfa Atrium

The Atrium of TECFA with its spring fountain is the place where TECFA people can relax and talk about the latest advances in Educational Technologies. Type a ROOM NUMBER to enter an room.

+

You see Machine a Cafe and WWW-box here.

>look www

WWW-box

A laptop sized WWW display. Type help 'www-box' or 'help www' !

'reset www' will reset - 'pick <n> on www' will display an item.

Remembered documents:

- [1] TECFA WWW home page
- [2] TECFA's WWW Home Page for Internal Use
- [3] Tecfa's page on educational MUDs
- [4] The TecfaMOO page
- [5] The TecfaMOO Manuel

Assignments

ASSIGNMENT 1

CREATE YOUR OWN MOO CHARACTER AND ROOMS

TecfaMOO

Connect

host: tecfamoo.unige.ch

port: 7777

First commands

After connecting via telnet or by your MOO client you'll receive the following:

```
<!--
000000 00000 00000 00000 00 M M MM MM Questions ?
  o  o  o  o  o  o MM MM M M M M MooMail: to kaspar
o  000  o  000 000000 M M M M M M M M
o  o  o  o  o  o M M M M M M M M
o  00000 00000 o  o  o M M MM MM
```

A Virtual Space for EDUCATIONAL TECHNOLOGY, EDUCATION, RESEARCH and LIFE at
TECFA, School of Psychology and Education, University of Geneva, Switzerland.

The TecfaMOO Project WWW page (including 'how to use' information):

<http://tecfa.unige.ch/tecfamoo.html>

Connect as a registered person by typing „connect (name) (password)“

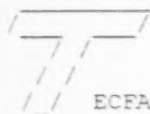
Guests: type 'connect guest' - Type 'who' to see who is connected.

>

* You can type 'connect guest'; you'll receive:

```
<!-- Connected --!>
```

TecfaMOO ARRIVAL area



Builders and Programmers Permits
are available to qualified people
ECFA MOO Arrival Area

A Virtual Space for Educational Technology, Education, Research & Life
at TECFA, School of Psychology & Education, University of Geneva

See: <http://tecfa.unige.ch/tecfamoo.html> for information on this project

'VISIT' (for Visitor Information) 'BUILD' (Builder's Information)

'Register' (User creation)

* ESF/LHM --> at the Educational Technology Center ('enter bus')

Code is copyrighted TECFA unless otherwise stated (e.g. ported code).

Non profit organizations can port things if they ask us.

You see Big Brother, NAVETTE SPECIALE, and CONFERENCE SHUTTLE here. Klimt (asleep), bino [ne pas deranger] (asleep), Bohli (asleep), and Dexa (dozing) are here.

Obvious Exits: Up (to Rond Point de Plainpalais (west section)), UC (to Underground Corridor), VISIT (to Visitor Information), TRANS (to Rond Point de Plainpalais (east section) / Taxi), BUILD (to Underground Construction Office), Register (to TecfaMOO Registration Room), and MUSEE (to Musee d'arts graphiques).

Current time in Geneva is : 11:53 a.m.

[Please enter your name:]

* You must type your name without blank character; you'll receive:

My_name

#\$#mcp version: „2.1“ to: „2.1“

There is new activity on the following list:

*new (#123) 52 new messages

There are new news items for you to read. Please type „news“ to get a summary.

* You can type 'who' to see who is connected: you'll receive:

User name	Connected	Idle time	Location
-----	-----	-----	-----
My_name[guest]	9 minutes	0 seconds	TecfaMOO ARRIVAL area
Vivian [alert]	an hour	7 seconds	Stafl3 Classroom
cathy [ok]	3 hours	8 seconds	Stafl3 Classroom
isa	2 hours	24 seconds	Stafl3 Classroom
hercule [Radu]	34 minutes	33 seconds	Detecting
DanielP [goguenar]	3 hours	a minute	Stafl3 Classroom
Lienad	37 minutes	18 minutes	Stafl3 Classroom
monitor [cecoco]	30 minutes	29 minutes	Mystery House: Foyer
boreal	2 hours	43 minutes	Phil's Place
pierre	an hour	50 minutes	Pierre's Office
Colin	21 hours	53 minutes	Colin's Office
Dexa	an hour	an hour	TecfaMOO ARRIVAL area
Kaspar [test]	8 days	an hour	Croisement Rue de Candolle/Cons
Dao	8 days	2 hours	The Laboratory
Thora	a day	2 hours	Secret Valley

Total: 15 users, 6 of whom have been active recently.

SchoolNet**Connect**

host: moo.schoolnet.ca

port: 7777

First commands

After the telnet or MOO client connect you'll receive the following:

This world is Pueblo 2.01 enhanced.

```

88b      d88      ,ad888ba,      ,ad888ba,
888b      d888      d8"      "8b      d8"      "8b
88`8b      d8'88      d8'      `8b      d8'      `8b
88  `8b d8' 88 88      88 88      88 88      88
88  `888' 88 88      88 88      88 88      88
88  `8' 88  Y8,      ,8P  Y8,      ,8P
88      88      Y8a.  .a8P  Y8a.  .a8P
88      88      "Y888Y"      "Y888Y"      Canada, Eh?

```

Telnet: moo.schoolnet.ca 7777 ### Telnet: moo.rescol.ca 7777

MOO Canada (SchoolNet MOO) is moving on Nov. 15th. If the above address does not work, try '24.64.89.15 7777'.

MOO Canada (MOO de Rescol) se deplace le 15 novembre. Si l'address ci-dessus ne fonctionne pas, essayer '24.64.89.15 7777'.

If you have a player, type: connect [name]

Si vous avez deja un personnage: accede [nom]

If you want to visit, type: connect guest

Si vous voulez simplement visiter: accede visiteur

To make a new player, type: request

Pour creer un nouveau peronnage: demande

* You can type 'connect guest'; you'll receive:

Welcome to the SchoolNet MOO. The MOO is a safe and secure environment for people of all ages to meet, talk, program, and do a dazzling array of other things. However, in order to keep this environment safe and best for everyone involved, we ask that you abide by the following general rules:

- * You may not swear.
- * You may not harrass anyone in any fashion.
- * You may not abuse anyone.
- * Everyone is entitled to respect, privacy, and the right to be here when they follow these rules.
- * And never play with the wizzen in an attempt to annoy them.

These are just the general rules. In order to stay here on the MOO, you must, must, read the policies in depth by typing 'help policy'.

Ignorance, failure to read these policies, is not an excuse for not following them. You will be given a warning before being removed from the MOO for a short time, and then permanently afterwards if you still don't learn.

-- The Wizzen.

Do you understand and promise to abide by these policies? [Enter 'yes' or 'no']

There is new news. Type 'news' for a summary.

* You must type 'yes'; you'll receive:

BEST COPY AVAILABLE

*** Connected ***

The Grassy Knoll [DT]

The grassy knoll is located in the centre of the down town area. You can hear the light traffic on Megabyte road in the distance.

You see a welcome sign, a flag pole, InterMaze Project, death, Tricorder Project, Slater's World (Sci-Fi), SandCrawler Project,

DownTown Restaurant Project, Heavy-Gear MOO, Dark, Dragon Tails Mountains, Eto, MetaChat Project, and QuakeII MOO here.

To the northeast, you see Schoolbook Depository [DT].

To the northwest, you see Help Center [DT].

To the south, you see The forest [DT].

To the west, you see Megabyte Road [2].

To the east, you see Megabyte road [1].

To the north, you see Vector Road.

Hello, you appear to be new to SchoolNet MOO. If you need help, you may type:
help me

at any time, and you will be instantly moved to the Help Centre where there are several helpful items.

* You can type 'who' to see who is connected: you'll receive:

Name	Connected	Idle	Location
-----	-----	-----	-----
Biology_Guest	6 minutes	0 seconds	The Grassy Knoll [DT]
bombone	24 seconds	24 seconds	My recreation room

Character Request

...at TecfaMOO

You should connect as guest and after that you should type 'request_character' - exercise!

...at SchoolNet

You should type 'request' just after you are connected with the MOO client (you should not be connected as any character).

Creating a new MOO account for you...

Please type in your e-mail address. This is where the password for your new account will be sent. It will be kept strictly confidential.

E-MAIL >

* You should type your E-mail address; you'll receive:

What would you like your new MOO character to be called?

You can rename your character later if you change your mind.

NAME >

* You should type your character name; you'll receive:

Request completed; <YOUR CHARACTER CHOICE> (#3777) created.
Sending password via e-mail...
Email sent off to <YOUR E-MAIL ADDRESS> successfully.
Check you mail, it has probably already arrived.
If you do not receive it within an hour, send email to MOO@schoolnet.ca!

*** Getting your right to build**

Normally you have right to build your own room and another rooms and/or objects.
The strategy is to create your home with the following command:
@dig „My_home“

After that move into the new created room with the following command:

@move me to #n
where #n is the identifier for your created room.

After that you can set your home there:

@sethome here

When you reconnect to the MOO you'll be in that room.

*** Getting your programming rights**

The programming right is easy to get at SchoolNet (you must go into some room and read something) - exercise!

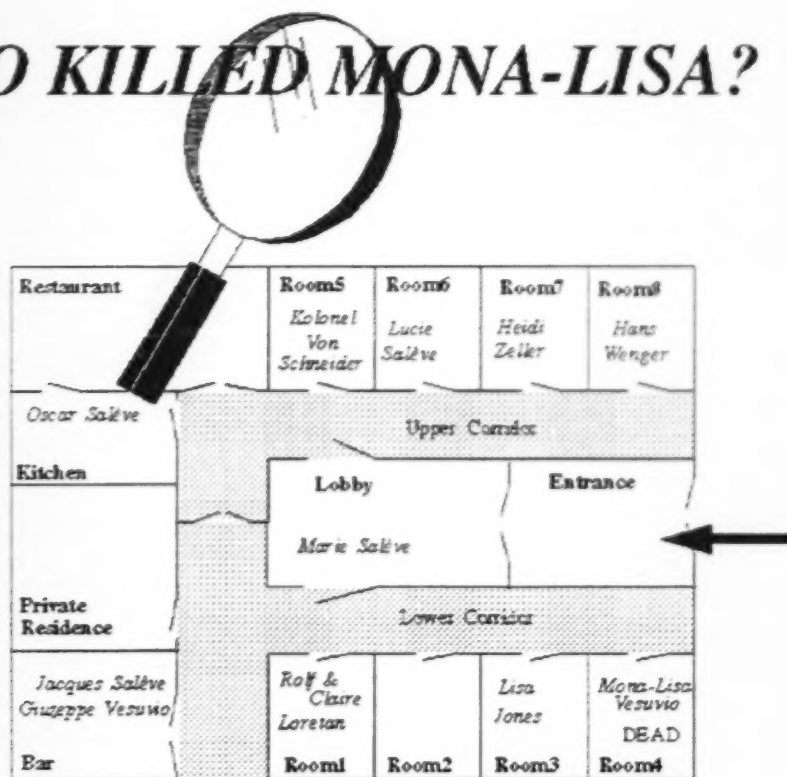
At TECFAMOO the right to programming must be obtained directly from the MOO wizard.

ASSIGNMENT 2**MOO MYSTERY GAME***Who killed MONA-LISA*

A MOO is a virtual environment on the network where people can meet and collaborate on various projects and tasks. The MOO mystery game described below was designed on TecfaMOO, a virtual MOO world at the University of Geneva. The purpose behind this game is to study collaborative problem solving in a virtual environment. This would allow to develop computational models for supporting human-computer collaboration (Dillenbourg, 1996). The specific goal of this experiment was to study how two human subjects elaborate a shared solution of a problem and how the drawings they make on a shared whiteboard help to support mutual understanding.

In the experiments, the two subjects play detectives (named Hercule and Sherlock) in a mystery solving game: Mona-Lisa has been killed and they have to find the killer.

WHO KILLED MONA-LISA?



Auberge du Bout de Nappe

Chez Marie et Oscar Salève

Mona Lisa Vesuvio was found dead by her husband at 10:30 PM in room 4. The body is still there. She was last seen in public at 9 PM when she left the restaurant. You must discover a suspect who has both a motive (e.g., money, jealousy) and the means and opportunity to kill the victim. For opportunity, locate the murder weapon and discover who had access to it at the time of the murder.

The human detectives walk in the MOO environment where they meet suspects, ask questions about relations with the victim, regarding what they have done the night of the murder, and so forth.

Suspects are programmed robots. When exploring rooms, they find various objects which help them to discover the murderer. They are told that they have to find the single suspect who (1) has a motive to kill, (2) had access to the murder weapon and (3) had the opportunity to kill the victim when she was alone. The murder has not been accomplished by an alliance between two or more people. Most people will be truthful, although, of course, the murderer may try to hide his or her own involvement. The map where the crime was happened is shown in figure below:

Please work with your partner as a team: share information and deductions, and try to reach a shared opinion as to who did it. The subjects are located in different places, connected by a computer network and communicate using two pieces of software: a text-based MOO system and a White-board. The MOO environment is a standard MOO called TECFAMOO. The subjects move in different rooms, talk to each other via two commands: „say...“, to communicate with anybody in the same room, and „page John...“ to communicate with John where ever he is.

The detectives each carry a notebook which automatically records the answer to all the questions that they ask. In this experiment, the subjects use a MOO client called TKMOOlight which runs on UNIX stations. It includes an elementary white-board: both users draw on a same page, can see and edit the objects drawn by their partner, but they do not see each other's cursor. All actions and interactions in the MOO window and in the white-board are recorded.

Basic MOO Commands

Dialogue

talk to somebody in same room (public)
 e.g. > say hi
 or > „Hi
 talk to somebody in another room (private)
 e.g. > page hercule where are you?
 or > 'hercule where are you?
 or > ' where are you? (to page last person paged)

Navigation

type look to see exits and then type the exit name
 e.g. > red
 see who is where
 e.g. > who

Action

look at objects.
 e.g. > look desk
 read a note
 e.g. > read note3

Advanced MOO Commands

Note: These will not work for normal TECFAMOO characters.

Interrogation

ask a suspect (in the same room) about last night
 e.g. > ask helmut about last night
 ask a suspect (in the same room) about the victim (Mona-Lisa Vesuvio)
 e.g. > ask hans about the victim
 ask a suspect (in the same room) about any object
 e.g. > ask giuseppe about jacket

Navigation

join somebody (a suspect or your partner) wherever they are (crossing intermediate rooms).

e.g. > join Lucie

go to a room whether or not it is connected by an exit (crossing intermediate rooms)

e.g. > walk to bar

Memory

read the answers from a suspect or all suspects, or previously read notes, these answers are stored in the detective notebooks (DN1 for Sherlock and DN2 for Hercule)

e.g. > read oscar from dn

e.g. > read all from dn

e.g. > read sign from dn

Coordination

Merge your notebook answers with your partner's notebook (so each of you can see all answers)

e.g. > compare dn1 with dn2

show or read an object to your partner

e.g. > show jacket to Hercule

e.g. > read note to hercule

follow your partner every time he leaves the room

e.g. > follow hercule

e.g. > dont_follow hercule (to stop)

forces your partner to follow you

e.g. > lead sherlock

e.g. > dont_lead sherlock (to stop)

How to connect to the game?

You should have:

1. a MOO client or a minimal telnet connection
2. a password for each detective character (cata@cib.pub.ro)

ASSIGNMENT 3
CHATTERBOTS

Get on the Web and „chat“ with a couple of chatter bots. (Some links below).

Describe in a paragraph what each of them was like as a conversational partner. Compare at least two of them, in terms of how human-like they are. Could any of these things convince you they were human? Does „chatting“ with these things give you more or less faith in the Turing test as a test of intelligence?

Chatterbot suggestions:

Dr. Webowitz

MegaHAL

Alice

Eliza, formatted for the Web

Eliza again

Eliza yet again

You will find the above mentioned bots as well as many others on:

<http://www.toptown.com/hp/sjlaven/>

ASSIGNMENT 4**SEMI-STRUCTURED INTERFACE FOR COLLABORATIVE PROBLEM SOLVING**

Run the mystery game by using the interface for collaboration with HercuLOG, the artificial Prolog agent.

The TCL/Tk plugin version 2.0 is required (you can download it from the following address: <http://www.seast2.usc.sun.com/960710/feature1/plugin.html>). You must allow trusted policy in your config file.

You have to follow the next steps:

1. Go to the <http://tecfa.unige.ch/~ceco/TCL/testb.html> Web page.
2. Put your (nick)name in the „user name box“ on the 2nd line. Press <Enter> after that.
3. Get connected to the MOO mystery game by pushing the MOO_Connect button on the 2nd line (you can see what's happened in the Sherlock's screen on the 6th line)
4. If fail try again later. STOP.
5. Login as Sherlock by pushing „Login“ button on the 2nd line.
6. If you fail (that's means that Sherlock is already log on), try again later. GO TO 15.
7. Call Hercule by pushing „Call Hercule“ button on the 1st line. If the Hercule is alive the „Call Hercule“ button is disabled and the „Begin“ button is enabled. If not, you can try again later and GO TO 15. To start Hercule agent you must follow the running Hercule paragraph.
8. Push the „Begin“ button on the 1st line to begin the mystery.
9. You can issue the commands:
 - a. the structured commands from the 4th line (you can see the command on the 3rd line and send with the „Send“ button)
 - b. any command from the 5th line (you can see the command on the 3rd line and send with the „Send“ button)
10. You can see all things of Sherlock on the 6th line.
11. You can say/ask something to Hercule:
 - a. the questions/facts/suggestions from the 7th line combined with the 8th line's box editor (you can send those things with the „Page“ button or assert from yourself with the „OK“ button on the 7th line)
 - b. the questions at the questions buttons on the 10th line from the sentence's head from the 9th line
12. You can see specific Hercule's answer(s) on the 11th line by choosing the head of the sentence on the 7th line or by choosing the Hercule's head of the sentence and the Hercule's answer(s) at the head of the sentence on the 9th line.
13. You can see all Hercule's things on the 12th line.
14. You can push the face to
 - a. break Hercule if the face smiles and the criminal is unknown
 - b. restart Hercule after break if the face is upset
 - c. end Hercule if the face smiles and the criminal is known (i.e. the mystery was solved)
15. You can exit from the Interface by pushing the „Exit“ button on the 2nd line. If you choose „yes“ from dialog box that you'll see the Interface will disappear. END!

References